A Data Dissemination Strategy in SDN Enabled Vehicular Networks

Chen Chen(✉), Na Li, Yansong Li, Ronghui Hou, and Zhiyuan Ren

State Key Laboratory of Integrated Service Networks,
Xidian University, Xi'an 710071, China
cc2000@mail.xidian.edu.cn

Abstract. In a vehicular network, the vehicles generally need to handoff among RSUs (Road Side Units) frequently on highways due to their rapid mobility and the limit of RSU's radio coverage. This issue may cause a series of problems such as data transmission interruption or increasing of the transmission delay. In this paper, we took advantages of the emerging idea of SDN to improve the performance of the vehicular networks. Specifically, we proposed a SDN-based framework for the vehicular networks. In this framework, we developed three function modules over the SDN application layer. And then by installing appropriate rules on OpenFlow enabled RSUs, the controller can execute a wise scheduling of RSUs' downlink streams. In addition, based on this framework, we proposed a data dissemination strategy when a vehicle handoff among the RSUs to reduce the latency especially for bulk traffic. Simulation results demonstrate that our solution can significantly reduce the latency and the retransmission rate. In the paper, we adopted classical DCF mechanism in the IEEE 802.11p standard to implement our protocol, which makes our solution practical and compatible with previous drafts.

Keywords: SDN · OpenFlow · Vehicular network · DCF · RSU handoff

1 Introduction

Nowadays, vehicular networks have attracted more and more attentions, which is regarded as the key technology to improve the road safety and the construction of intelligent transportation system [1]. In a vehicular network, the communication between vehicles is known as V2V (vehicle-to-vehicle), and the vehicle and the infrastructure such as the RSU (roadside unit) is called V2I (vehicle-to-infrastructure). The V2V communication usually has strong randomness due to high dynamic of vehicles, therefore V2I communication is supposed to play an important role to improve the performance of the network especially in high mobility scenarios. In RSU-aided networks, vehicles traveling along the road can establish stable connections to RSUs deployed on the roadsides, which is able to provide high-bandwidth communication capability for data transmissions [2].

However, in traditional vehicular networks, the server always puts all the content in one RSU, while the coverage of existing RSU is still limited. As a result, the vehicle in the high-speed mobile scenarios such as highways will access another RSU even it

Q. Chen et al. (Eds.): ChinaCom 2016, Part II, LNICST 210, pp. 302–313, 2018.
DOI: 10.1007/978-3-319-66628-0_29

doesn't take off all the contents from the previous attached RSU. When the vehicle successfully access the new RSU, it has to send the content requirement again [3]. As we can see, this process above causes much unnecessary communication overhead, especially transferring the bulk data and needing to switch RSUs frequently.

To solve the problems above, in this paper we propose a novel SDN based VANETs architecture and a kind of data dissemination strategy during the RSU handoff. The core idea of SDN [4, 5] is to realize the flexible controlling of the network traffic by decoupling the control plane and the data plane of the network equipment. SDN changes the original network with hardware configuration as the core to a software based network, which can greatly improve the performance in centralized controlling, programmability and other aspects of the existing architecture. In this paper, we first construct the SDN based VENETs architecture and explain the functions of every part in detail. Then through derivation we cut the bulk data the vehicle requested into different sizes and allocate to the corresponding RSU. Furthermore, we propose the pre-cache mechanism so that can reduce the kinds of communication overhead at utmost. At the end of the paper, we simulate the architecture and dissemination strategy we proposed in OPENET to verify the effectiveness of our solution.

2 Background and Related Work

2.1 Distributed Channel Access Mechanism in 802.11

So far the WLAN based 802.11p/wave protocol [6] is an important part of vehicle networks. The Federal Communications Commission (FCC) appoints 75 MHz band for vehicle networks communication based 802.11p. Among them, the first 1 MHz band is reserved as a security vacuum boundary, the rest is divided into seven adjacent 10 MHz band. As shown in Fig. 1, the channel is sequentially numbered Ch172, Ch174,…, Ch184, in which Ch178 is controlling channel CCH for controlling and managing the other six bands, the Ch172 is using as emergency message transmission channel, and the Ch184 is suitable for transmitting long-distance public safety information, at the same time, Ch174, Ch176, Ch180, Ch182 are common traffic channels using to transfer the traditional network data through RSUs [7].

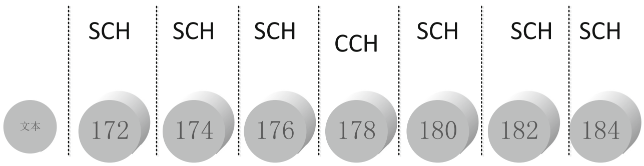

Fig. 1. Channel allocation of IEEE 802.11p Mac Layer

As known, common traffic channels have characteristics of traffic burst, low delay requirement and high throughput requirement which coincide with the CSMA/DCF protocol. Therefore, we consider using CSMA/DCF technology to compete ordinary

SCH channel slot in this paper. The vehicle passes messages occupying this slot to the RSU by one or two hops. Through the OpenFlow switches the messages in RSU are transferred to SDN controller, and then injected into the Internet. In the downlink, the messages from the Internet are sent to the RSU through the SDN controller, then the RSU completes the downlink slot forwarding to the vehicle by one hop or two hops at most.

2.2 Related Work

With the developing of SDN, the domestic and foreign scholars have conducted a lot of works about applying SDN in different fields. The network architecture based SDN proposed in a REF. [8] which has explained the function of each part in detail. More importantly, the practicability of the architecture is illustrated by the example of lane change prediction and traffic flow distribution at the last part of this paper. REF. [9] leveraged the emerging idea of SDN and OpenFlow technology to reorganize the architecture of enterprise WLAN to mitigate the impact of interference, which cannot be handled very well in conventional architecture. They proposed a downlink packets scheduling algorithm to mitigate the impact of interference among APs and clients, as well as conducted fine-grained downlink packets scheduling by installing appropriate rules in corresponding APs.

However, the solution about reducing kinds of communication overhead during the vehicle handoff process is still approaching the traditional method. The server will put all the data that the vehicle requested in RSUs along the way the vehicle passing. This solution will cause storage wasting and connection delay. So far few scholars propose a directive solution, yet their researches on other aspects have provided important references for us. REF. [10] proposed a hybrid Vehicle networks architecture aimed to improve the protocol extensibility which is similar with mesh network. The paper studied how to deploy RSU based vehicle traffic conditions and urban road structures. However its research scene is only in one simple high-way, leading the deployment issue is simplified as determining the access point distance according to the vehicle density. Liu [11] have analyzed the RSU flow of uplink and downlink, as well as put forward to take vehicles on the road as relay nodes for data transmission.

3 Data Dissemination During Handoffs Among RSUs

3.1 Framework and Handling Process

The vehicle network architecture based SDN is shown in the Fig. 2. For the convenience of research we select the highway no considering the export and turning, besides that, the RSU blind spots and the overlapping caused by more than two RSUs are out of consideration. Among them, RUS is a roadside unit with the function that can collect information including the road length, road conditions, the number of vehicles and location and state of nearby RSUs etc. At the same time, RSU can also communicate with the OBU in vehicle, collecting and storing the vehicle information including vehicle ID, speed, route and the requirement etc. Finally RSU as a relay device is responsible to forward the information above and information from the controller [12].

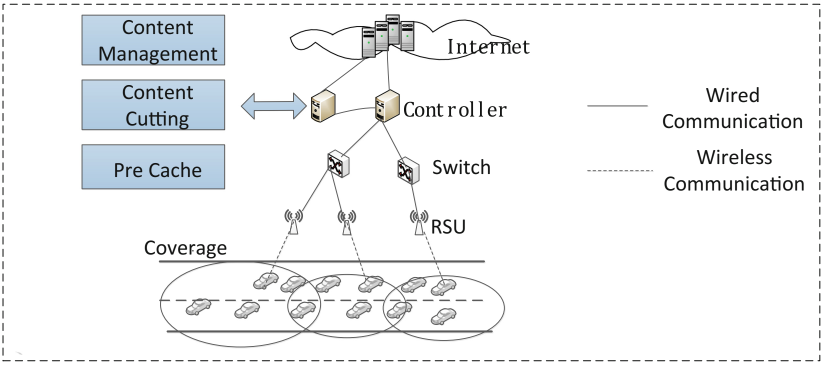

Fig. 2. The vehicle networks architecture based SDN

Each OpenFlow switch [13] in architecture is composed of three parts: security channels, OpenFlow protocols and flow tables. Security channel is for the communication between the switch and the controller transmitting the protocol commands and data packets. The OpenFlow protocol is the standard interface for communication between the controller and the OpenFlow switch. Each flow table contains a number of flow table entries, each of which contains a matching field and an action set, according which the switch processes the flow.

We select Floodlight as the controller that is responsible for the development and management of the forwarding strategy. Floodlight [14] uses a modular architecture to achieve the corresponding functions, which has a good extensibility. In this paper, through the controller north interface (API) we develop three modules respectively: Content Management, Content Cutting and Pre Cache. The Content Management module includes data management and user management. Data management is mainly used to manage the server IP, the location of RSUs, the current network topology and the data has been cached. User management records vehicle ID, speed, route, as well as the requirement. The Content Cutting module divides the data into different segments according to the information in Content Management module. Pre Cache module sends the packets to the corresponding RSU through the appropriate transmission path.

The top layer is Internet, which contains a large number of original servers, which is the terminal of data upload and download. The V2I communication adopts 802.11p protocol, while the connection of other parts adopts wired connection to ensure the reliability of transmission.

The System operation is shown in Fig. 3.

(1) The vehicle sends the vehicle information and request content to RSU through the OBU in vehicle.
(2) RSU as a relay device sends the information above to the SDN controller through the switches.
(3) Content Management module is triggered to update user information and data information.

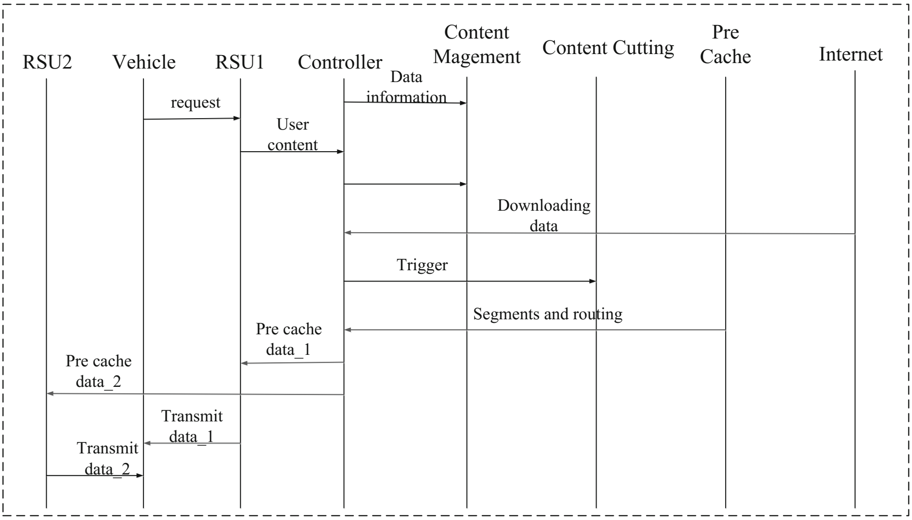

Fig. 3. System operation process

(4) The SDN controller interacts with the Internet interaction uploading or downloading the data.
(5) Content Cutting module divides the data received from Internet into different segments and packs them in unified form.
(6) Pre Cache module sends the packets to the corresponding RSU through the appropriate route.
(7) When the vehicle enters the corresponding RSU, the RSU authenticates the vehicle and transmits the packets pre cached in it.

3.2 Content Management

Since the SDN controller needs to make the routing strategies of global information, it is necessary to keep track of all the information and topology of the current network. In this paper, we use the Content Management module to store the data content and user information. The data content includes RSU management and data management.

The RSU manages packets as form in Table 1. The RSUs along the road are numbered as RSU_A, RSU_B, RSU_C, etc. The Location is just the location of the RSU. Name represents the name of the data that has been cached in the RSU. The throughput in the Table 1 is given when the vehicle is in the overlapping of two RSUs and they all have the content requested, then we will select the larger throughput of RSU for transmission (Here we ignore the RSU overlapping situation due to space limitations, and it is another research point). Data management includes the size of the packet, the update time, and the original server. Its format is shown in Table 2. User information packets include the user's ID, location, vehicle speed, travel path, and the requirement. The frame format is shown in Table 3.

Table 1. RSU management packet

Identity	Location	Name	Throughput
RSU_A	{31.7811, 230.0018}	Data1	320 Mb
RSU_B	{31.1368, 230.9945}	Data2	512 Mb
RSU_C	{31.0025, 230.8597}	Data3	128 Mb

Table 2. Data management packet

Identity	Size	Update time	Owner
Data1	1024	201601220530	{A, C}
Data2	800	201502243122	{B}

Table 3. User management packet

Identity	Location	Speed	Travel path	Requirement
Vehicle_1	{31.2564, 230.1269}	20 m/s	{D, E, G}	Data 1
Vehicle_2	{24.2356, 230.5612}	15 m/s	{D, E, F}	Data 3

3.3 Content Cutting

IEEE 802.11p protocol currently supports the maximum transmission distance within 1000 m, which can meet the communication requirement between the vehicles driving at 33 m/s (about 120 km/h) and between the vehicles and RSUs. Considering the constraints of antenna transmitting power, we suppose that the RSU transmission radius is Rr = 1000 m, and the vehicle density in the RSU is ρ, thus the number of vehicles is $\lambda = \rho 2R_r$. As known, the number of vehicles obeys Poisson distribution [15] so that the expectation and variance are λ. The probability function of the number of vehicles in one RSU is:

$$\rho(n) = \frac{\lambda^n e^{-\lambda}}{n!} = \frac{\rho 2R_r)^n e^{-\rho 2R_r}}{n!}. \tag{1}$$

In the coverage range of the RSU, assuming the probability of a vehicle sending a data frame in a random time slot is τ, then the probability of collision between this data frame and other data frame sending from another node is. When the value of n is given, η will remain dependent and constant. The relationship is as following:

$$\eta = 1 - (1 - \tau)^{\mathrm{n}-1}. \tag{2}$$

$$\tau = 1 - (1 - \eta)^{\mathrm{n}-1}. \tag{3}$$

Bianchi analyzes the DCF backoff process using Markov chain. Given W = CWmin (minimum contention window), maximum contention window CWmax = 2 m−1, m is the retransmission rate. According to the analysis, The probability of sending a data frame when n vehicles compete the wireless channel is:

$$\tau(\eta) = \frac{2}{1 + W + \eta W \sum_{i=0}^{m-1} (2\eta)^i}. \tag{4}$$

From formula (3) and (4), we can work out the value of and. It is assumed that the probability of at least one frame in a time slot transmitting is P_{tr}, and only one frame transmitted successfully is P_s

$$P_{tr} = 1 - (1 - \tau)^n \tag{5}$$

$$P_s = \frac{n\tau(1-\tau)^{n-1}}{P_{tr}} = \frac{n\tau(1-\tau)^{n-1}}{1-(1-\tau)^{n-1}} \tag{6}$$

If a vehicle node successfully transmits a frame, the channel is busy for other nodes. The probability of this process is $P_{st} = P_{tr}P_s$, the average time is T_{st}; When the channel is idle, the conflict probability that a node sends data during this time is $P_{co} = P_{tr}(1 - P_s)$, the average time is T_{co}; The idle channel probability during a time slot is $P_{idle} = (1 - P_s)$, the average time is T_σ Among them, $P_{st} = T_{DIFS} + T_{DATA} + T_{SIFS} + T_{ACK}$,

$P_{co} = T_{DIFS} + T_{CODATA} + T_{SIFS} + T_{ACK} \cdot T_{SIFS}$ is minimum frame interval in DCF mechanism; T_{DIFS} is the waiting time between the idle channel confirmation and sending data; T_{ACK} is the acknowledgement frame length; T_{DATA} is a frame data transmission time; T_{CODATA} is the maximum data frame length when conflict occurs. If we assume that all the data frame length is same, then $P_{st} = P_{co}$. The average time of a system time slot is:

$$\begin{aligned} T_{slot} &= P_{idle}T_\sigma + P_{st}T_{st} + P_{co}T_{co} \\ &= (1-\tau)^n T_\sigma + (1-(1-\tau)^n)(T_{DIFS} + T_{DATA} + T_{SIFS} + T_{ACK}) \end{aligned} \tag{7}$$

The transmission efficiency is:

$$\begin{aligned} \Psi(n) &= \frac{P_{tr}P_sT_{DATA}}{P_{idle}T_\sigma + P_{st}T_{st} + P_{co}T_{co}} \\ &= \frac{n\tau(1-\tau)^{n-1}T_{DATA}}{(1-\tau)^nT_\sigma + (1-(1-\tau)^n)(T_{DIFS} + T_{DATA} + T_{SIFS} + T_{ACK})} \end{aligned} \tag{8}$$

According to formula (8), if there is D_{DATA} needed to transmit, the traffic that can successfully transmitted during a frame is $D_{SDATA} = \Psi(n)D_{DATA}$. In 802.11p protocol, the data transmission rate $\omega = \frac{D_{DATA}}{T_{DATA}}$.

It is analyzed in REF. [5], if there are n vehicles in the coverage of the RSU, the total throughput is:

$$\begin{aligned} E(\Gamma) &= E[\textstyle\sum_{n=0}^{\infty}(np(n)DS_{DATA})\frac{\omega}{n}\frac{1}{D_{DATA}}] \\ &= E[\textstyle\sum_{n=0}^{\infty}(n\frac{(\rho 2R_r)^n e^{-\rho 2R_r}}{n!}\psi(n)D_{DATA})\frac{\omega}{n}\frac{1}{D_{DATA}}] \\ &= E[\textstyle\sum_{n=0}^{\infty}(\frac{(\rho 2R_r)^n e^{-\rho 2R_r}}{n!}\psi(n))]\omega \end{aligned} \tag{9}$$

Therefore, the probability that there is one vehicle at least in RSU according to the function (1) is $1 - e^{-\rho 2R_r}$. It is also the vehicle i existing probability. And the vehicle i downloading time is:$\Delta t = \frac{2R_r}{v_i}$.

Thus, the total traffic that vehicle i directly downloads (one hop) from RSU is:

$$E[W] = E[\Gamma_{downlink}\Delta t] \approx \frac{(1 - e^{-2\rho R_r})\omega T_{DATA}}{(\frac{2W-1}{2})T_\sigma + (T_{DIFS} + T_{DATA} + T_{SIFS} + T_{ACK})} \cdot \frac{2R_r}{v_i} \tag{10}$$

If the vehicle i only downloads within the distance d, the downloading traffic is $E(W_d)$:

$$E[W] = E[\Gamma_{downlink}\Delta t] \approx \frac{(1 - e^{-2\rho R_r})\omega T_{DATA}}{(\frac{2W-1}{2})T_\sigma + (T_{DIFS} + T_{DATA} + T_{SIFS} + T_{ACK})} \cdot \frac{d}{v_i} \tag{11}$$

Assuming the total traffic that vehicle i requested is $E(Q)$, if $E(Q) < E(W_d)$, there is no need to switch RSU completing the download. If $E(Q) > E(W_d)$, it is necessary to switch RSU, it means that we need to cut the download data into different packets. We suppose the traffic downloading from RSU_A is $E(W_{d1})$, if $E(Q) - E(W_{d1}) > E(W)$, we allocate a size of $E(W)$ packet into RSU_B. Otherwise, put the rest of data into RSU_B totally. Using this solution, cut the data sequentially and divide the packet into corresponding RSU. The segmented data is packed as Table 4.

Table 4. Segmented data packet

Name	Original IP	Segments	Destination	Size
Data_1	10.0.0.1.5	Data_1.1	RSU_A	$E(W_{d1})$
		Data_1.2	RSU_B	$E(W)$
		Data_1.3	RSU_C	The rest

3.4 Pre Cache

In order to allow the vehicle to connect with the RSU as soon as possible and reduce the latency maximally, we adopt the pre-cache pattern. The PRE_CACHE packet is shown in Table 5.

We name the time that the vehicle gets to the next RSU as Time to Live (TTL). Flag is a number of Boolean, which 1 represents downloaded already, and 0 is not downloaded yet. Continue Cache represents whether other vehicles requests the data either. When there are multiple vehicles request the same data, we can use the Continue Cache value represents the arrival time of the vehicle requested later. The value of TTL and Continue Cache will decreases over time. If the TTL expires (TTL < 0) and

Table 5. PRE_CACHE packet

Name	Original server IP	TTL	Flag	Continue cache
Data_1.1	{10.0.0.9}	80	1	100
Data_3.2.1	{10.0.0.12}	100	0	0
Data_5.4.1	{10.0.0.20}	−50	1	30

Continue Cache value is not 0, then the TTL is replaced by the value of Continue Cache. For example, the first column of the Table indicates that the vehicle requesting the Data_1.1 will come in 80 s; the original server IP of the data is 10.0.0.9 and the data has already cached in this RSU. At the same time, another vehicle requesting the data ether will come in 100 s. Besides that, the TTL value will be added to the Continue Cache value when the RSU receives the packet and finds that it has been cached already.

3.5 Authentication and Handoff

When RSU detects a vehicle in its coverage, the algorithm below is activated immediately. Given the vehicle ID set in RSU is ϕ, and the ID of vehicle i is ID_k. Then we use the pseudo code in the Table 6 to authenticate the vehicle.

Table 6. Matching algorithm

Matching algorithm
Input: ID_k
1. Switch (ID_k), if case i == ID_k, return case i, default.
2. End switch.

Once the vehicle i has been found through the Matching Algorithm, the RSU will send the packets stored in it for vehicle i to it. Thus, the whole dissemination process during the handoff is completed.

4 Performance Evaluation

In this paper, we use OPNET simulator to build the SDN enabled vehicular networks architecture. The simulation architecture and parameters we set are shown in Fig. 4 and Table 7 respectively. Based on this architecture, we verify the advantages of the proposed data dissemination strategy by latency and retransmission ratio.

We first compared the average transmission delay in SDN-based framework with data dissemination strategy against that in traditional framework. In order to verify the strategy performance during the handoff among RSUs we select to transmit the bulk traffic packets trough RSUs to the vehicle, which varies from 0M to 500M. The results are given in Fig. 5. SDN-based framework with the proposed dissemination strategy can reduce the transmission delay by at least 30%. Then we vary the transmission traffic so that the vehicle needs to through more than two RSUs to complete the

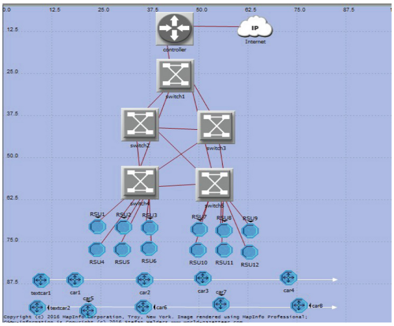

Fig. 4. Simulation model

Table 7. Simulation parameter

Parameter	Value
Mac	802.11p
Wired network bandwidth	20 Mbps
RSU wireless bandwidth	10 Mbps
RSUs spacing and coverage	1000 m
Vehicle speed	20–25 m/s

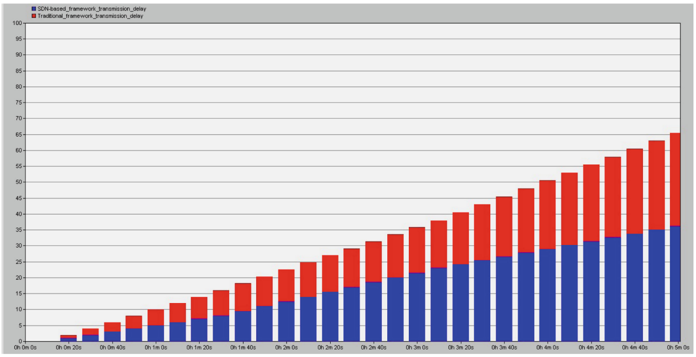

Fig. 5. Transmission delay

transmission, we found that when the offered traffic is low, the improvement is not quite obvious. With the increase of offered traffic, the disparity between SDN-based framework and the traditional framework is more and more obvious. This means that our solution will be more effective when vehicle needs to switch RSU during the transmission.

We further evaluated the retransmission ratio in Fig. 6. As we can see in traditional framework the retransmissions ratio increases rapidly as the increase of packets traffic. On the contrary, the ratio in our solution keeps at about 0.10, The phenomenon indicates that our dissemination strategy can avoid the interruption and reduce the reconnection time, as a result the retransmission ratio can be maintained in a lower level.

At the last, we evaluated the RSU pre cache success ratio. As we known, in traditional VENETs data dissemination mechanism, the bulk data will always be cached totally, which may occupy a lot of memory space in RSU. Thus will cause the cache failure when data is cached to RSU. Our plan will solve this problem perfectly, cause the bulk data will be cut to different segments and then pre cached to corresponding RSUs. In Fig. 7, we can see under traditional data dissemination mechanism, the cache success ratio is about 0.2 to 0.4, while using our solution, the cache success ratio can reach 0.8 to 0.95. It has increased substantially.

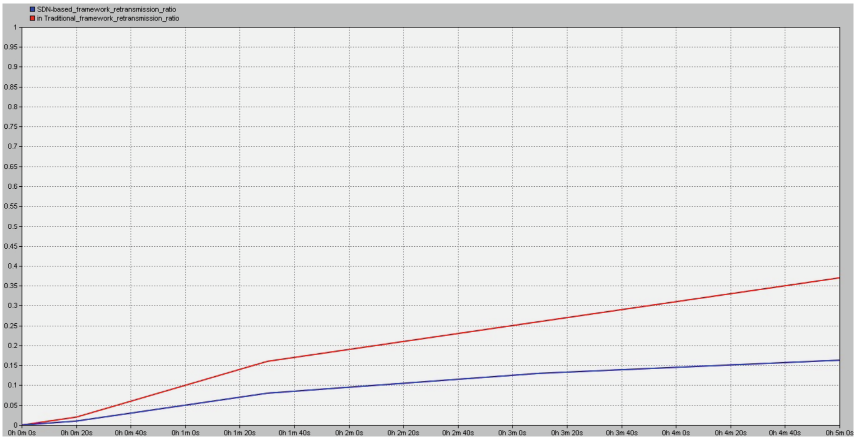

Fig. 6. Retransmission ratio

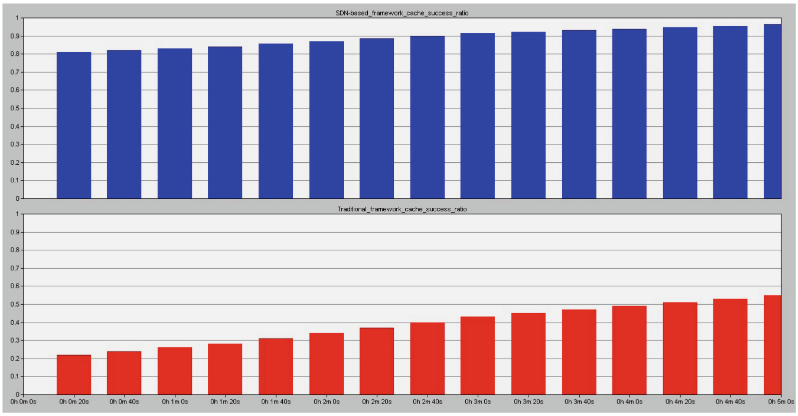

Fig. 7. Cache success ratio

5 Conclusion

The paper based on the traditional traffic distribution theory and classical CSMA/DCF mechanism in 802.11p protocol, constructed a new vehicle network architecture based SDN. More importantly, we propose a data dissemination strategy during the RSU handoff, which is elaborated in detail in the paper. At the last, we simulated the dissemination strategy in OPENET, as expected, the simulation shows that the strategy based the architecture we proposed can reduce the latency and retransmission rate significantly. The following work may carry research on the multi- vehicle competition situation.

Acknowledgments. This work was supported by the National Natural Science Foundation of China (61201133, 61571338 and 61672131), the key research and development plan of Shaanxi province (2017ZDCXL-GY-05-01), the National Key Research and Development Program of China (2016YFE0123000), the National Science and Technology Major Project of the Ministry of Science and Technology of China (2015zx03002006-003, MJ-2014-S-37), the Natural Science Foundation of Shaanxi Province (2014JM2-6089), the National High-tech R&D Program of China (863 Program-2015AA015701), the Ningbo Huimin projects of science and technology (2015C50047), the Research collaboration innovation program of Xi'an (CXY1522-3) and the "111 Project" of China (B08038).

References

1. Toor, Y., Muhlethaler, P., Laouiti, A.: Vehicle ad hoc networks: applications and related technical issues. In: IEEE Communications Surveys and Tutorials, vol. 10, pp. 74–88. IEEE Press (2008)
2. Trullols-Cruces, O., Morillo-Pozo, J., Barcelo, J.M., Garcia-Vidal, J.: A cooperative vehicular network framework. In: IEEE International Conference on Communications, pp. 1–6. IEEE Press (2009)
3. Giuseppe, B.: Performance analysis of the IEEE 802.11 distributed coordination function. IEEE J. Select. Areas Commun. **18**, 535–547 (2000)
4. ONF White Paper: Software-Defined Networking: The New Norm for Networks. Open Networking Foundation (ONF) (2012)
5. Bernardos, C.J., de la Oliva, A., Serrano, P.: An architecture for software defined wireless networking. J. IEEE Wirel. Commun. **21**(3), 52–61 (2014)
6. Piscataway, N.: IEEE standard for information technology-local and metropolitan area networks-specific requirements—Part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications, amendment 6: wireless access in vehicular environments, pp. 1–51 (2010)
7. Labudde, P.: IEEE standard for wireless access in vehicular environments (WAVE). J. Multi-channel Oper. **1609**, 4–7 (2005)
8. Truong, N.B., Lee, G.M., Ghamri-Doudane, Y.: Software defined networking-based vehicular ad hoc network with fog computing. In: IEEE International Symposium on Integrated Network Management (2015)
9. Zhao, D., Zhu, M., Xu, M.: Leveraging SDN and OpenFlow to mitigate interference in enterprise WLAN. J. Netw. **9**, 1526–1533 (2014)
10. Kaisser, F., Veque, V.: On the scalability problem of high-way ad hoc network. In: IEEE Wireless Communications and Networking Conference, Istanbul (2009)
11. Liu, Y.: The research of RSU downlink traffic in vehicle networks. J. Softw. (2015). (in Chinese)
12. Ku, I., Lu, Y., Cerqueira, E., Gomes, R., Gerla., M.: Towards software-defined VANET: architectures and services. In: Mediterranean Ad Hoc Networking Workshop, pp. 103–110. IEEE Press (2014)
13. OpenFlow Switch. http://openvswitch.org/
14. Floodlight. http://www.projectfloodlight.org/floodlight/
15. Gerlough, D.L.: Poisson and traffic: use of Poisson distribution in highway traffic. Distributions (1990)