DOI: 10.3969/j. issn. 1673-5188. 2017. 02. 002
http://kns.cnki.net/kcms/detail/34.1294.TN.20170426.1121.002.html, published online April 26, 2017

Software Defined Networking Based On-Demand Routing Protocol in Vehicle Ad-Hoc Networks

DONG Baihong, WU Weigang, YANG Zhiwei, and LI Junjie
(Guangdong Province Key Lab. of Big Data Analysis and Processing, School of Data and Computer Science, Sun Yat-Sen University, Guangzhou 510006, China)

Abstract

This paper comes up with a SDN Based Vehicle Ad-Hoc On-Demand Routing Protocol (SVAO), which separates the data forwarding layer and network control layer, as in software defined networking (SDN), to enhance data transmission efficiency within vehicle ad-hoc networks (VANETs). The roadside service unit plays the role of local controller and is in charge of selecting vehicles to forward packets within a road segment. All the vehicles state in the road. Correspondingly, a two-level design is used. The global level is distributed and adopts a ranked query scheme to collect vehicle information and determine the road segments along which a message should be forwarded. On the other hand, the local level is in charge of selecting forwarding vehicles in each road segment determined by the global level. We implement two routing algorithms of SVAO, and compare their performance in our simulation. We compare SVAO with popular ad-hoc network routing protocols, including Optimized Link State Routing (OLSR), Dynamic Source Routing (DSR), Destination Sequence Distance Vector (DSDV), and distance-based routing protocol (DB) via simulations. We consider the impact of vehicle density, speed on data transmission rate and average packet delay. The simulation results show that SVAO performs better than the others in large-scale networks or with high vehicle speeds.

Keywords

VANETs; SDN; routing protocol; ad-hoc network; Internet of Vehicle

1 Introduction

Nowadays, with the popularity of personal vehicles, transportation systems are facing many problems, including traffic congestion, environment pollution, increasing energy consumption, etc. [1]. Therefore, the research of intelligent transportation system (ITS) has become a magnet for researchers in recent years, in which vehicle ad-hoc networks (VANETs) play a key role. In VANETs, a vehicle can either communicate with another vehicle directly via Vehicle-to-Vehicle (V2V) communications, or communicate to infrastructure such as a road side unit (RSU) via Vehicle-to-Infrastructure (V2I) links [2]. As an intersection of traffic network and traditional ad-hoc network, VANETs hold some of their features, for example, it has decentralized, self-organizing and dynamic topology. VANETs also have unique features in aspect of structure, support high-mobility scenarios and implement special traffic applications.

This research is partially supported by National Key Research and Development Program of China (2016YFB0200400), National Natural Science Foundation of China (No. 61379157), Program of Science and Technology of Guangdong (No. 2015B010111001), and MOE-CMCC Joint Research Fund of China ((No. MCM20160104).

Message/packet routing and forwarding have been always a core issue for VANETs. How to find, build and choose an appropriate route in a highly dynamic vehicle network can be a tough problem, especially with stability, packet delay, computation overhead, and bandwidth taken into account.

Quite a lot of work has been conducted on designing routing protocols and forwarding mechanisms for VANETs. Considering the information required for routing protocols, the current routing work in VANETs can be concluded as follow: topology based protocols, position based protocols, map based protocols, and road based protocols. Topology based protocols route and forward according to the topology of road segments, no matter whether there exists a global route table such as Destination Sequence Distance Vector (DSDV) [3], or a local one such as Dynamic Source Routing (DSR) [4] and Ad-hoc On-demand Distance Vector Routing (AODV) [5]. Position based protocols only care for the position of vehicles such as Greedy Perimeter Coordinator Routing (GPCR) [6], or the position of RSUs such as Intersection-based Geographical Routing Protocol (IGRP) [7]. Map based protocols attempt to take some road segment states into account. Geographic Source Routing (GSR) [8] considers about the crossroad, while Shortest-Path-Based Traffic-Light-Aware Routing (STAR) [9] concerns the impact of the traffic light. Road based protocols focus on the communication among road segments such as Vehicle-Assisted Data Delivery (VADD) [10]. However, most of the researches above mainly focus on distributed routing discovery and selection process. In reality, due to the difficulty of coordinating among the nodes, the data transmission efficiency is limited.

Software defined networking (SDN) is a promising technology aiming to reinvent the Internet and improve the network transmission efficiency. It separates the data forward layer and the network control layer, realizing a more effectively centralized network resource allocation and scheduling. It hammers at making the network more intelligent and realizing reasonable network resource utilization. In a traditional network, the network control layer and data forward layer are tight coupling on the routing unit, which makes it hard for applying the latest network control technology on it. So the traditional routing unit is so-called "simple" and "dumb" [11]. It is also referred to as "Internet ossification" [12]. On the one hand, researchers have to take various hardware differences between routers into account when they want to apply a new routing protocol. On the other hand, when hardware needs to be updated, whether to support the operation of the existing routing protocol should be considered as well. That is what SDN focuses on. SDN separates the logic control layer and data forward layer, so as to spearate the route unit and control unit. In detail, the routing unit is responsible for forwarding according to the flow table. The generation, maintenance, configuration of flow table is managed by the control unit. The separation of control layer and data forward layer realizes flexible forward strategy management of SDN system.

However, as SDN is originally designed for general cable wide area networks (WANs) and local area networks (LANs), the routing and forwarding mechanism need to be redesigned when we try to apply SDN into VANETs. A VANET involves wireless sensor network, wireless cellular network, mobile self-organizing network, and more. Furthermore, the node mobility, architecture, and traffic rule effect of the VANET make it more complicated than the traditional wireless network.

This paper comes up with a SDN based on-demand routing mechanism under the on-demand VANETs forwarding scenario, aiming at improving the routing and forwarding efficiency. The largest challenge of this paper is redesigning routing and forwarding mechanism in VANETs. This paper proposes a two-level structure protocol, SDN Based Vehicle Ad-Hoc On-demand Routing Protocol (SVAO), including the distributed global level consisting of local controllers (LCs) and the centralized local level consisting of vehicles. The global level is responsible for calculating and distributing the route among road segments/LCs, while the local level in charge of calculating the route for each vehicle in each section. We implement two routing algorithms of SVAO in this paper, Bellman-Ford algorithm and Floyd algorithm, to explore the difference of their performance. Finally, we compare the performance of the new protocol with four traditional self-organizing network routing protocols (Optimized Link State Routing (OLSR), DSR, DSDV and distance-based routing protocol (DB)). The simulation result shows that the proposed protocol performs better.

2 Related Work

The routing protocol is vital for VANETs, whose features include highly dynamic topology, wide coverage and low robustness. A suitable routing protocol can keep up a good transmission quality in VANETs. According to different types of needed information, the current routing protocols in VANETs can be divided into the following four categories: topology based protocols, position based protocols, map based protocols, and road based protocols.

2.1 Topology Based Protocols

DSDV [3] is a table-driven scheme based on Bellman-Ford algorithm. Each entry in the route table contains a sequence number, which identifies whether this entry is valid. The sequence number is generated by the destination node, and periodically updated by sending packets. Routing information is updated by sending packets among nodes periodically and triggered incremental updates packets more frequently.

DSR [4] is a source routing type on-demand routing protocol. DSR is beacons-less, which means that the update of the network topology information will not rely on periodically sending Hello packet (beacons). The main process of establishing a route described as follow: at the very beginning, the source node floods the RouteRequest packet to the whole network; once the destination node receives a RouteRequest packet, it will return a RouteReply packet along the reverse path; lastly the route tables alongside the path are updated by adding the route from the source node to the destination node.

AODV [5] floods route messages to conduct route discovery, similar to DSR. The difference between them is that AODV is a node routing scheme, that is, when a node (whether it is the destination node or not) receives a RouteRequest packet, and this node has cached the latest route towards the destination node, it will not forward this packet but return a RouteReply packet along the reverse path and update the route table alongside the path.

Due to the high-mobility and highly dynamic network topology in VANETs, the main difficulty of topology-based protocols is how to reduce the route discovery cost, time cost and resource cost. If the time cost is high, the latest route will lose efficacy frequently.

2.2 Position Based Protocols

GPCR [6] is a greedy routing protocol without link state. Different from traditional routing protocols, nodes directly take use of the position information of their neighbors to establish a route instead of finding the shortest path towards destination node or judging the reachability of the route. The source node always greedily sends packets towards the node that is most close to the destination node, until it could not reach the destination node. Then it will begin a new round of forwarding and the greedy strategy will be adopted in a new node again, until the packet reaches a closer node towards the destination node.

IGRP [7] is designed to solve the QoS routing problems under the city traffic scenario. Since it may in fact not be feasible and economical to absolutely deploy RSUs, IGRP attempts to ensure the maximum packet received rate by changing the point-to-point routing strategy to check whether the RSUs exist.

2.3 Map Based Protocols

GSR [8] is a routing protocol based on map and vehicle position. It uses Dijkstra algorithm to compute the shortest path from the source node towards the destination node. Each crossroad acts as an anchor node, and a greedy forwarding strategy is adopted between each two anchor nodes. With a high data packet reception rate, GSR is adaptive to the situation that contains more vehicles and larger traffic density. It does not suitable for the condition with less car and poor connectivity.

STAR [9] takes the impact of the traffic light into account. It regards the traffic light as a main factor in the dynamic network topology in the practical city traffic scenario. The state of the traffic light in crossroad will be treated as an influence factor in the packet forwarding strategy. In practice, STAR has lower average delay, higher packet reception rate and higher TCP throughput capacity in unit time.

2.4 Road Based Protocols

VADD [10] adopts a store-and-forward strategy which is adaptive to the scenario with low vehicle density and poor connectivity especially. Predicting the future position of the vehicle, every packet will be marked with a packet transfer delay, which ensures the vehicle can store the packet until the vehicle reaches next road.

The distributed routing protocols in current vehicle networks aim at taking the balance among route computation overhead, link hops, link quality and link stability. Due to the dynamic nature of VANETs, a distributed protocol may cause large link state delay to discover a route. To settle the problems above, this paper proposes an SDN based on-demand forwarding routing protocol which adds local controller as a roadside unit in current VANETs and gets use of the advantages of centralized method to get an optimized routing mechanism.

Recently, there are already some works on SDN in VANETs. S.-A. Lazar and C.-E. Stefan concluded that SDN and fog working together can be the best solution to VANETs, after overviewing both side of works [13]. K. Liu and J. K. Y. Ng were the first focusing on scheduling cooperative data dissemination in hybrid infrastructure-to-vehicle (I2V) and vehicle-to-vehicle (V2V) communication environments [14]. A. Kazmi and M. A. Khan tried to deal with problems in topology dynamics and connectivity losses. They exploited SDN planes by partitioning VANETs to work in distributed manner, which is better than traditional VANETs architectures [15]. X. Xiao and X. Kui analyzed the traces of taxies in Shanghai to find out the regular pattern of the inter-contact time between moving vehicles and intersections, and the duration of each contact, making way for the future study of SDN VANETs [16]. A. Di Maio and M. R. Palattella focused on the security issues for enabling SDN in VANETS in real use cases (smart parking, smart grid of electric vehicles, platooning, and emergency services) [17]. A. KazmiEmail and M. A. Khan kept their eyes on single point of failure (SPOF) in SDN, coming up with an abstracted VANET model but also complying with SDN principals [18].

3 System Model and Introduced Coefficient

In this section, we firstly describe the system model and then introduce the coefficient used in proposed protocol.

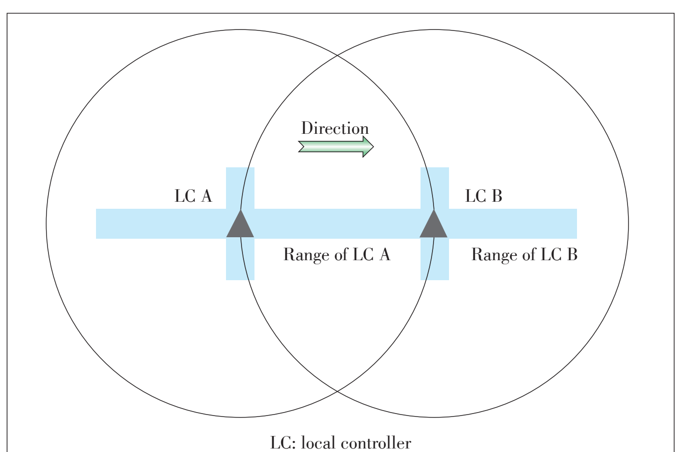
3.1 Sytem Model

Firstly, the city road is simplified into a neat network diagram. We assume that every vehicle moves towards the same direction in each road. That is, this paper only considers one-way road. A roadside control unit named LC will be placed in every crossroad (**Fig. 1**), which is responsible for collecting all the information of the vehicles on this road and computing route. LCs can communicate with each other to exchange information about road segment states.

3.2 Link Stability Coefficient

In VANETs, the transmission quality and efficiency are influenced by many factors, such as the hop number of the link, the relative speed of the vehicles in the link, etc. For the sake of quantitative evaluation, we propose a link stability coefficient (CV) to evaluate the stability of different link. The following notions are used in our formulas.

- n is the hop count of link between two adjacent LC,



▲**Figure 1. The positions of LC.**

- *sd* is the standard variance of the vehicle speed on the link,
- *av* is the average velocity of all the car in the road,
- V_i is the velocity of the car *i*.

The formula of CV is shown as follow. (1) and (2) take the relative velocity and node number of link into account. (3) refers to the coefficient of variation in math, ensuring the CVs of different road links with different node numbers can be compared immediately.

$$sd=\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(V_i-aV\right)^2}, \tag{1}$$

$$av=\frac{\sum V_i}{n}, \tag{2}$$

$$CV=\frac{sd}{av}*100\%. \tag{3}$$

4 Two-Level Structure Protocol Design

Based on the system model, we propose a two-level SDN based Vehicle Ad-Hoc On-Demand Routing Protocol (SVAO), with a global level and a local level. The global level is distributed, using the ranked query scheme to query the objective vehicle information and the improved AODV method to calculate the route among multiple road segments/LCs. The local level is centralized, using Bellman-Ford algorithm and Floyd algorithm to maintain a stable route between two adjacent LCs. That is, to explore the different impacts of the routing algorithms, this paper implements two type of shortest path algorithms for computing route.

4.1 Local Level

The local level is responsible for computing the route for every vehicle on each road. Firstly, we need to maintain a stable vehicle link between two adjacent LCs. Every vehicle will flood Hello Message (IDv, GPS, Speed) to LC periodically. When a vehicle enters a road segment, its Hello Message will be collected by the LC. Within a certain time, LC will collect all the hello information from the vehicles on the road, and try to build a network topology for the road. Since the Hello Message contains the velocity information of the vehicle, LC can predict the vehicle movement trajectory. Furthermore, LC will simulate and predict the topology changes according to the network topology we build above. At last, LC try to maintain a shortest transmission path from this LC towards the next adjacent LC, using Bellman-Ford algorithm (**Algorithm 1**) and Floyd algorithm (**Algorithm 2**).

Algorithm 1. The Bellman-Ford Algorithm

- *T* := road network topology
- **for each** vehicle *i* **do**//initialization
- *Distant[i]*=10000;
- *Distant[s]*=0;
- **end for**
- **for each** edge(*u,v*) in *T* **do**
- **if** edge(*u,v*) == 1 **then**
- *w*(*u,v*) =1;
- **else** *w*(*u,v*) =0;
- **end for**
- **boot** neg =false;
- **for each** nodes *u* in *T* **do** //relaxation
- **for each** nodes *v* in *T* **do**
- **if** *Distant[u]* +*w*(*u,v*) < *Distant[v]* **then**
- *Distant[v]* = *Distant[u]* + *w*(*u,v*)
- neg = true;
- **end for**
- **end for**
- **if** neg == false **then**
- **end**;
- **else** send error;

Algorithm 2. The Floyd Algorithm

- *T* := road network topology
- *D*:= the minimum distances between any two vehicle
- **for each** vehicle *i* **do**//initialization
- **for each** vehicle *j* **do**
- *D[i][j]=infinity;*
- **end for**
- **end for**
- **for each** vertex *v*
- *D[v][v]*=0;
- **end for**
- **for each** edge(*u,v*) in *T* **do**
- **if** edge(*u,v*) exist **then**
- *D[u][v]*=1;
- **end for**
- **for** *k* **from** 1 **to** size(*T*) **do** //floyd
- **for** *k* **from** 1 **to** size(*T*) **do**
- **for** *k* **from** 1 **to** size(*T*) **do**
- **if** *D[u][v]* > *D[u][k]* + *D[k][v]* **then**
- *D*[*u*][*v*] = *D*[*u*][*k*] + *D*[*k*][*j*];
- **end for**
- **end for**
- **end for**

4.2 Global Level

The global level is responsible for finding the position of the objective vehicle and for calculating the global route among road segments/LCs. We assume that vehicle S needs to send a message to the vehicle D. Firstly, vehicle S sends a RouteRequest message towards LC, asking for the position of D. If LC finds the location of D, it will return a route table to S, indicating the next hop.

4.2.1 Ranked Query Strategy

Once a LC receives a RouteRequest packet, it will begin to query for the position of vehicle D. In order to reduce the query cost, LC uses a ranked query strategy like cellular network. That is, LC is divided into two levels: the first level LC caches all the vehicle information in the road segment under its jurisdiction (Vehicle_ID, Direction, Speed), and then the second level LC only caches the vector of LC ID and vehicle ID instead (Vehicle _ID, LcID). The ratio between the first level LC and second level LC should be 20:1.

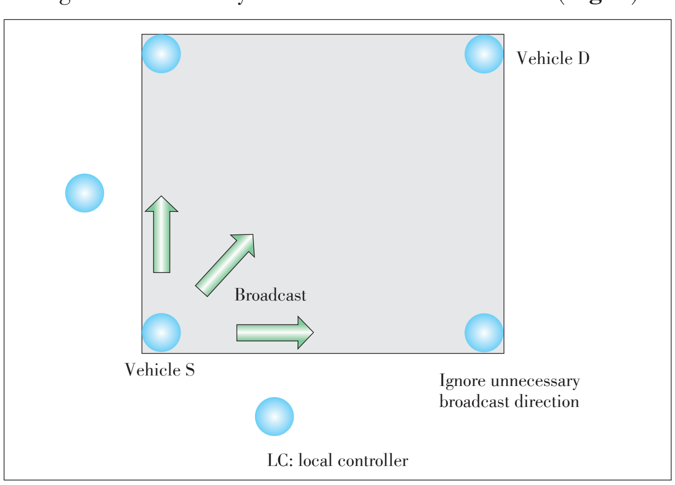
The ranked query strategy process is as follows:

1) Fisrtly, LC queries whether vehicle D is within the scope of its signal coverage.
2) If not, LC will send the query request to its second level LC. The second level LC then queries for the information of vehicle D among all the first level LCs within its signal coverage.
3) If not, the second level LC will flood the query request within all the second level LCs.
4) The process above is repeated until the position of vehicle D is found.

4.2.2 Computing Global Route Using Improved AODV Algorithm

Until now, each LC has maintained a network topology for corresponding road segment. Then, we need to compute a global route among road segments/LCs, connecting source vehicle S and destination vehicle D. We use the optimized AODV algorithm to get the route running in LC. The algorithm is shown as follow:

1) LC broadcasts a RREQ (Route Request) packet to other LCs.
2) To reduce broadcast scale and constrain RREQ broadcast direction, LC checks whether the next hop is wihtin the rectangle constituted by source S and destination D (**Fig. 2**). If not, LC drops the packet.
3) LC counts the passed LC number N from source vehicle S to itself. N will be compared and the best one in smaller CV_max will be selected; If N_new > N_his, we just drop the packet.
4) Within a certain time, the destination LC has received a series of RREQ packets. To make a best decision, LC tends to select the route with the smallest N, and the smallest CV_max takes the second place in the case of the same N. Then the destination LC sends a RREP (Route Request Respond) packet following in reverse route taken by the RREQ packet, updating the routing table of all the passed LCs.
5) LC adds the first vehicle information of the next road segement/LC into the route table, and broadcast the route table to all the vehicles within its jurisdiction.



▲**Figure 2. LC ignores unnecessary broadcast directions.**

5 Simulation Results and Analysis

For the sake of cost and safety, it is impossible to setup experiments with a practical network topology, which needs a large number of cars. So researchers usually test their solutions via simulation. We also setup experiments in NS-3, a famous simulation platform, and use Simulation of Urban MObility (SUMO) to generate vehicle trajectory files.

5.1 Simulation Setup

In order to evaluate the performance of the proposed routing protocol, we compare SVAO with OLSR, DSR, DSDV and DB within different vehicle density, at different vehicle speeds and in different communication ranges. Moreover, as this paper implement two different route computing algorithm, Bellman - Ford and Floyd, there are two different versions of SVAO in the following figures, while SVAO - BF represents the SVAO version using Bellman - Ford and SVAO - Flo represents the Floyd version. Data transmission and average packet delay are measured to get the contrast ratio. The simulation specifications with NS-3 are shown in **Table 1**.

We use map software SUMO to generate a three-lane road, whose length is 2 km. The road is uniformly divided into two sections. One section ranges from 0 km to 1 km and the other from 1 km to 2 km. At the very beginning of the simulation, the road is empty. Then the vehicles begin to appear in a random starting point at the beginning of the first section, moving forward to the second section at a certain speed. This paper sets vehicle S on the roadside that is 5.1 m away from the beginning of the road as the source vehicle and vehicle D on the end of the road as the destination vehicle. Vehicle S will send packets to vehicle D constantly.

5.2 Impact of Vehicle Density on Protocol Performance

The data transmission rate and average packet delay are important indicators to evaluate the performance of a routing protocol. We compare all the routing protocols quantitatively by

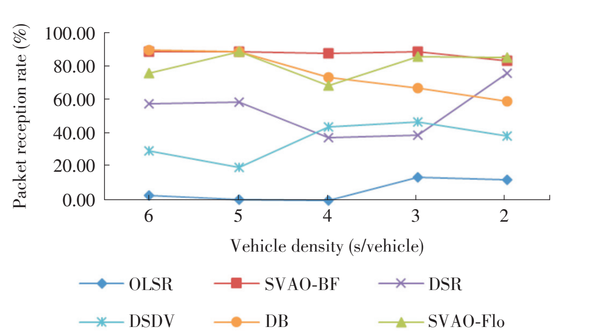
▼Table 1. Simulation setup

Parameter	Simulation value
SCH transmission radius (m)	400
CCH transmission radius (m)	1000
Vehicle speed (m/s)	10, 15, 20, 25, 30
Node movement mode	Random
Packet size (byte)	1000
Routing protocol	SVAO-BF, OLSR, SVAO-Flo, DSR, DSDV, DB
Road number	2
Road length (m)	1000

BF: Bellman-Ford Algorithm
CCH: control channel
DB: distance-based routing protocol
DSDV: Destination Sequence Distance Vector
DSR: Dynamic Source Routing
Flo: Floyd Algorithm
OLSR: Optimized Link State Routing
SCH: service channel
SVAO: SDN Based Vehicle Ad-Hoc On-Demand Routing Protocol

changing the density of nodes in the network. **Fig. 3** shows the packet reception rate under different vehicle densities. The average speed of each vehicle is 15 m/s. The horizontal axis shows that there will be a new vehicle appear in the very beginning of the road every *n* seconds. Under the same conditions, SVAO holds a better data transmission rate than OLSR, DSDV, DSR and DB. Along with the increasing number of nodes, i.e. the network scales larger, OLSR performs stably with a low value, DSDV and DSR performs undulated, and DB performs worse and worse. On the other hand, with the network scale expanding, SVAO maintains a stable lower value. SVAO-BF performs more stable than SVAO-Flo. It is because the time complexity of Floyd algorithm is higher than Bellman-Ford. That means it will cost more time to compute route when the density is getting higher. Both of them hold a good performance in average packet delay with a high vehicle density.

As for average packet delay illustrated in **Fig. 4**, the superiority of SVAO is remarkable and intuitive. In general, simulation results show that, as the number of nodes grows, SVAO obtains better performance and is more adaptive to the large-scale network. When it comes to the average packet delay, the two algorithms of SVAO perform well and similar, because both of them can find a relatively shortest road for routing.

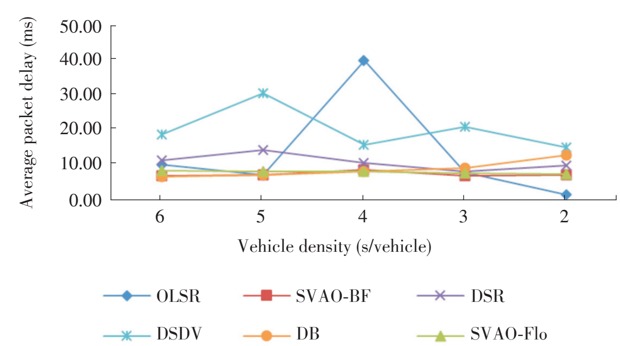


BF: Bellman-Ford Algorithm
DB: distance-based routing protocol
DSDV: Destination Sequence Distance Vector
DSR: Dynamic Source Routing
Flo: Floyd Algorithm
OLSR: Optimized Link State Routing
SVAO: SDN Based Vehicle Ad-Hoc On-Demand Routing Protocol

▲**Figure 3. Packet reception rates under different vehicle densities.**

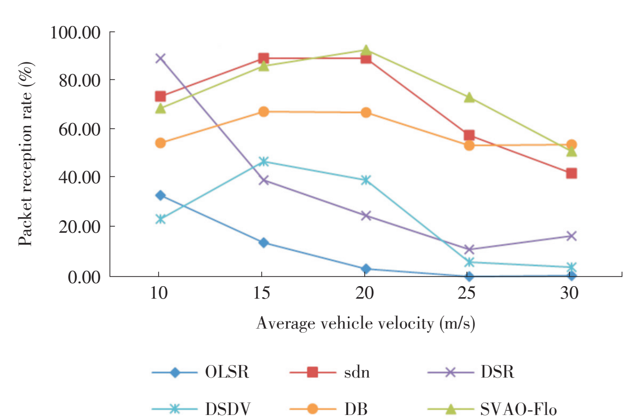
5.3 Impact of Vehicle Velocity on Protocol Performance

It can be seen from **Fig. 5** that at the same vehicle velocity, SVAO has an observably better packet reception rate. Along with the increase of speed, i.e. the road topology changes more frequently, all the protocols are observed a decline. The DSDV and DSR performance drop shapely, while the decline of SVAO and DB are unapparent. From the perspective of average packet delay (**Fig. 6**), with the vehicle speed gradually increasing, SVAO always keeps the best average packet delay (a lower value), while DSDV and OLSR perform worse. When it comes to the comparison between SVAO-BF and SVAO-Flo, we can see that they have the similar trend. When the vehicle goes faster, SVAO-Flo performs a little better. In short, simulation results show that with the vehicle speed increasing, i.e. the network topology of the road changes drastically, SVAO



BF: Bellman-Ford Algorithm
DB: distance-based routing protocol
DSDV: Destination Sequence Distance Vector
DSR: Dynamic Source Routing
Flo: Floyd Algorithm
OLSR: Optimized Link State Routing
SVAO: SDN Based Vehicle Ad-Hoc On-Demand Routing Protocol

▲**Figure 4. Average packet delay under different vehicle densities.**



BF: Bellman-Ford Algorithm
DB: distance-based routing protocol
DSDV: Destination Sequence Distance Vector
DSR: Dynamic Source Routing
Flo: Floyd Algorithm
OLSR: Optimized Link State Routing
SVAO: SDN Based Vehicle Ad-Hoc On-Demand Routing Protocol

▲**Figure 5. Packet reception rates under different vehicle velocities.**

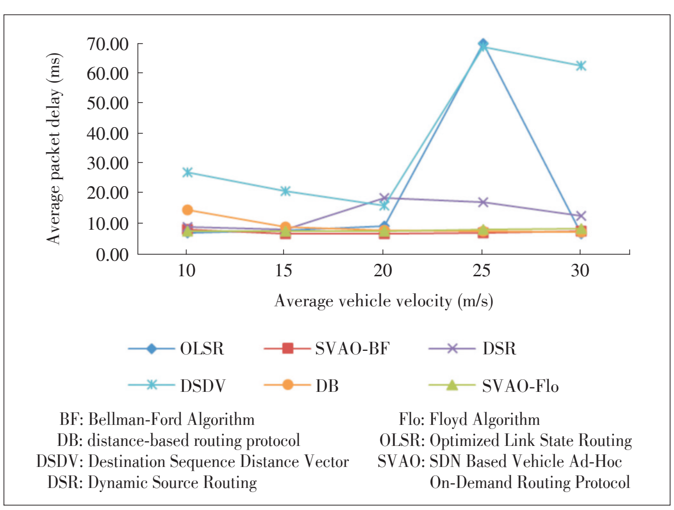
has a more stable and excellent packet reception rate and average packet delay.

5.4 Impact of Communication Range on Protocol Performance

Fig. 7 shows that SVAO has a better performance on packet reception rate under the same vehicle communication range. Along with the communication range increasing, the road topology becomes more complex, but the connectivity of road topology improves at the same time. In view of packet reception rate, the order is SVAO > DB > DSDV > DSR > OLSR. However, when the vehicle communication range becomes larger (larger than 400 m), DB performs better than SVAO. That is because DB can find a shorter path towards the destination easier although the topology becomes more complex. The average packet delay illustrated in **Fig. 8** is similar to sections 5.2 and 5.3, i.e. SVAO can obtain a stable and better performance. In general, along with the increase of the vehicle communication range, the road topology becomes more complex and SVAO performs better in both packet reception rate and average packet delay. Moreover, as shown in Fig. 7, the SVAO-Flo has a higher time complexity than SVAO-BF and the performance of SVAO-Flo can be better within a certain density and different communication ranges. That is, SVAO-Flo is more adaptable under different communication ranges.

5.5 Impact of Route Computation Interval on SVAO Performance

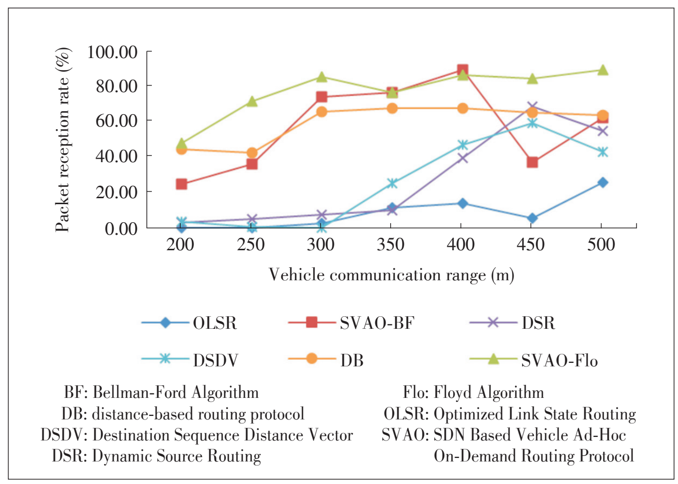
In this subsection, we collect data information from SVAO-BF with different route computing intervals. SVAO-BF adopts centralized network architecture, and routing discovery and computation process are conducted on the LC, which effectively reduces the route computing overhead. Therefore, how to achieve balance on the route computation cost and time cost is an important research direction. We quantitatively evaluate the impact of different route computation intervals on SVAO-BF performance. It can be seen from **Fig. 9** that a 10 s–14 s route computation interval can be a balanced choice. On the one hand, reducing the interval cannot improve the packet reception rate but can increase the route computing overhead. On the other hand, increasing the route computing interval will significantly reduces the SVAO-BF packet reception rate.



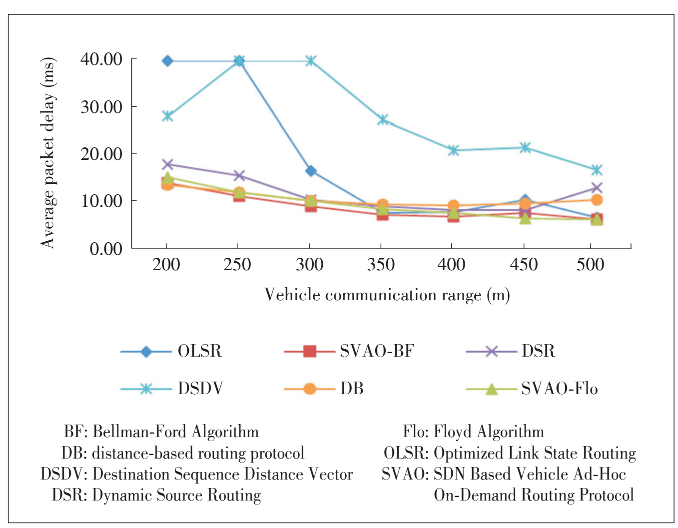
▲**Figure 6. Average packet delay under different vehicle velocitie**

6 Conclusions and Future Work

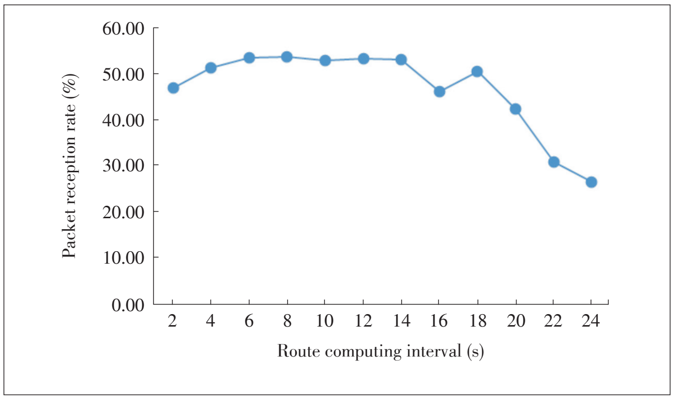
In this paper, we propose a SDN based on-demand routing protocol (SVAO), utilizing the separation of the data transfer layer and network control layer of SDN to enhance the data transmission efficiency within VANETs. Different from similar works on VANETs, our main work focuses on redesigning the network control layer and data transfer layer in VANETs, making SDN implemented in VANETs. This paper comes up with a two-level structure, including a distributed global level and a



▲**Figure 7. Packet reception rates under different vehicle communication ranges.**



▲**Figure 8. Average packet delay under different vehicle communication ranges.**



▲**Figure 9. Packet reception rates under different route computing intervals.**

centralized local level. The simulation results show that, in view of packet reception rate and average packet delay, SVAO performs better and more stable than traditional ad-hoc routing protocols (DSR, DSDV, OLSR and DB). In more detailed, SVAO-Flo has a similar performance with SVAO-BF no matter in average packer delay and packet reception rate, and SVAO-Flo is more adaptable under different vehicle communication ranges.

In future work, we may focus on the connection between two adjacent links. How to obtain the balance between the node number in a link and the stability of a link will be a promising direction. What's more, the simulation results show that SVAO performance will decline sharply under high-speed conditions. This needs further optimization.

References

[1] G. Samara, W. A. H. Al-Salihy, and R. Sures, "Security analysis of vehicular ad hoc nerworks (VANET)," in *2010 Second International Conference on Network Applications, Protocols and Services*, Alor Setar, Malaysia, 2010, pp. 55–60. doi: 10.1109/NETAPPS.2010.17.

[2] S. U. Rehman, M. A. Khan, T. A. Zia, and L. Zheng, "Vehicular ad-hoc networks (VANETs)—an overview and challenges," *EURASIP Journal on Wireless Communications and Networking*, vol. 3, no. 3, pp. 29–38, 2013. doi: 10.5923/j.jwnc.20130303.02.

[3] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," *ACM SIGCOMM Computer Communication Review*, vol. 24, no. 4, pp. 234–244, 1994. doi: 10.1145/190809.190336.

[4] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," *Mobile Computing*, vol. 353, pp. 153–181, 1996. doi: 10.1007/978-0-585-29603-6_5.

[5] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. 2nd IEEE Workshop on Mobile Computing Systems and Applications*, 1999, New Orleans, USA, pp. 90–100. doi: 10.1109/MCSA.1999.749281.

[6] C. Lochert, M. Mauve, H. Füßler, and H. Hartenstein, "Geographic routing in city scenarios," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 9, no. 1, p. 69, 2005. doi: 10.1145/1055959.1055970.

[7] H. Saleet, R. Langar, K. Naik, et al., "Intersection-based geographical routing protocol for VANETs: a proposal and analysis," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 9, pp. 4560–4574, 2011. doi: 10.1109/TVT.2011.2173510.

[8] C. Lochert, H. Hartenstein, J. Tian, et al., "A routing strategy for vehicular ad hoc networks in city environments," in *Proc. IEEE IV2003 Intelligent Vehicles Symposium, Columbus*, USA, pp. 156–161, 2003. doi: 10.1109/IVS.2003.1212901.

[9] J. J. Chang, Y. H. Li, W. Liao, and I. C. Chang, "Intersection-based routing for urban vehicular communications with traffic-light considerations," *IEEE Wireless Communications*, vol. 19, no. 1, pp. 82–88, 2012. doi: 10.1109/MWC.2012.6155880.

[10] J. Zhao and G. Cao, "VADD: vehicle-assisted data delivery in vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 3, pp. 1910–1922, 2008. doi: 10.1109/TVT.2007.901869.

[11] Q.-Y. Zuo, M. Chen, G.-S. Zhao, et al., "Research on openflow-based SDN technologies," *Journal of Software*, vol. 24, no. 5, pp. 1078–1097, 2013. doi: 10.3724/SP.J.1001.2013.04390.

[12] B. N. Astuto, X. N. Nguyen, K. Obraczka, T. Turletti, and M. Mendonca, "A survey of software defined networking : past, present, and future of programmable networks," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1617–1634, 2014. doi: 10.5923/j.jwnc.20130303.02.

[13] S.-A. L. Lazar and C.-E. Stefan, "Future vehicular networks: what control technologies?" in *International Conference on Communications*, Bucharest, Romania, 2016, pp. 337–340, 2016. doi: 10.1109/ICComm.2016.7528203.

[14] K. Liu, J. K. Y. Ng, V. C. S. Lee, S. H. Son, and I. Stojmenovic, "Cooperative data scheduling in hybrid vehicular ad hoc networks: VANET as a software defined network," *IEEE/ACM Transactions on Networking*, vol. 24, no. 3, pp. 1759–1773, 2016. doi: 10.1109/SURV.2014.012214.00180.

[15] A. Kazmi, M. A. Khan, and M. U. Akram, "DeVANET: decentralized software-defined VANET architecture," in *IEEE International Conference on Cloud Engineering*, Berlin, Germany, 2016, pp. 42–47. doi: 10.1109/IC2EW.2016.1662.

[16] X. F. Xiao and X. Kui, "The characterizes of communication contacts between vehicles and intersections for software-defined vehicular networks," *Mobile Networks and Applications*, vol. 20, no. 1, pp. 98–104, 2015. doi: 10.1007/s11036-014-0535-6.

[17] A. Di Maio, M. R. Palattella, R. Soua, et al., "Enabling SDN in VANETs: what is the impact on security?" *Sensors*, vol. 16, no. 12, p. 2077, 2016. doi: 10.3390/s16122077.

[18] A. Kazmi, M. A. Khan, F. Bashir, et al., "Model driven architecture for decentralized software defined VANETs," in *Future Intelligent Vehicular Technologies*, LNICST, vol. 185. Cham, Switzerland: Springer, 2017, pp. 46–56. doi: 10.1007/978-3-319-51207-5_5.

Manuscript received: 2017-01-20

Biographies

DONG Baihong (bh.dong@foxmail.com) received the B.Sc. degree in computer science from Sun Yat-sen University, China in 2012. He is studying for a M.Sc. degree in Sun Yat-sen University. His research interest is the SDN-based vehicular ad-hoc networks.

WU Weigang (wuweig@mail.sysu.edu.cn) received the B.Sc. degree in 1998 and the M.Sc. degree in 2003, both from Xi'an Jiaotong University, China. He received the Ph.D. degree in computer science in 2007 from Hong Kong Polytechnic University, China. He is currently an associate professor at the Department of Computer Science, Sun Yat-sen University, China. His research interests include distributed systems and wireless networks, especially cloud computing platforms and ad-hoc networks. He has published more than 50 papers in conferences and journals. He has served as a member of editorial board of two international journals, *Frontiers of Computer Science* and *Ad-Hoc & Sensor Wireless Networks*. He is also an organizing/program committee member for many international conferences. He is a member of the IEEE and ACM.

YANG Zhiwei (zhiwei200654@163.com) received the B.Sc. degree in 2006 and the M.Sc. degree in 2008, both from Sun Yat-sen University, China. He is currently a Ph.D. student majored in software and computer theory in Sun Yat-sen University. His research interests include vehicular ad-hoc networks and distributed systems, especially dynamic networks. He mainly focuses on information dissemination, counting, consensus and dynamic models in dynamic network, and has published related papers.

LI Junjie (goals.lee@qq.com) received the B.Sc. degree in automation science from South China University of Technology, China in 2012. He is studying for a M.Sc. degree in Sun Yat-sen University. His research interest is the SDN-based vehicular ad-hoc networks.