IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 31, 2018, accepted November 25, 2018, date of publication December 4, 2018,
date of current version December 27, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2884708

A Hierarchical Routing Scheme With Load Balancing in Software Defined Vehicular Ad Hoc Networks

YANGSHUI GAO[1,2], ZHILONG ZHANG[1,2], DAN ZHAO[1,2], YI ZHANG[3], AND TAO LUO[1,2], (Senior Member, IEEE)

[1]Beijing Laboratory of Advanced Information Networks, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2]Beijing Key Laboratory of Network System Architecture and Convergence, Beijing University of Posts and Telecommunications, Beijing 100876, China
[3]Shanghai Urban-Rural Construction and Transportation Development Research Institute, Shanghai 200032, China

Corresponding author: Tao Luo (tluo@bupt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61571065, in part by the 111 Project under Grant B17007, in part by the Beijing Natural Science Foundation under Grant L172032 and Grant L161005, and in part by the Fundamental Research Funds for the Central Universities under Grant 2017RC07.

ABSTRACT The vast majority of routing approaches for vehicular ad hoc networks (VANETs) are distributed, which are ineffective to exploit the global networking information and easily lead to local optimum, appearing as sparse connectivity and network congestion. Recently, by combining software-defined network architecture together with VANETs, software-defined VANETs (SDVN) is widely concerned, which shows the benefits of centralized control. In this paper, we propose a hierarchical geography routing protocol for SDVN. First, the protocol divides a large region into multiple small grids according to the geographical location and finds a series of grids with good connectivity based on real-time grid vehicle density and historical vehicle transfer probability between grids. Second, we construct a path cost function with load balancing and keep two paths with minimal costs from the selected grids. Finally, a series of relay nodes on each selected path are filtered for routing according to node utility. Simulation results show that the proposed routing algorithm achieves significant gains in terms of delivery ratio, throughput, average delay, and average hop count compared with several existing routing protocols.

INDEX TERMS VANETs, SDN, routing, load balancing, centralized control.

I. INTRODUCTION

Vehicle Ad hoc Networks (VANETs) have a large potential in boosting road safety and making commuting an enjoyable experience [1], [2]. Various types of information related to vehicle mobility on the road, such as vehicle density, speed, and directions of the vehicles as well as the weather, can be shared among vehicles via VANETs. This information helps to organize road traffic and prevent accidents [3]–[7].

Routing protocols which decide how a data packet is switched from source to destination are extremely important for the effective operation of the VANETs. According to whether routing protocols make use of geographic information, we classify them into two main categories: non-position based routing protocols [8], [9] and position based routing protocols [10]–[15]. For non-position based routing, all nodes store complete or partial information about the network topology. This incurs high maintenance cost and the topology information has to been frequently updated because of the mobility of vehicles. To overcome these drawbacks, position based routing is developed where the position information of neighbor nodes can be stored. Furtherly, depending on whether the centralized control mode is used, the position based routing protocols can be divided into non-centralized based [10]–[14] and centralized position based routing protocols [15]. Although the distributed algorithm achieves low complexity, it easily traps into local maximum, which may lead to sparse connectivity and network congestion. In contrast, by combining software defined network (SDN) [16] with VANETs, centralized routing algorithms are applicable, where the global network information can be obtained in real time, and the connectivity of the whole network is available for routing. These advantages allow the central controller to make better routing decisions.

Despite existing position based routing protocols provide various ways to improve the performance of VANETs, they generally fail to monitor the entire network in real time. Also, the network load is not considered in these protocols, which may risk network load imbalance. For instance, if the sending end vehicles (e.g. source vehicles or relay vehicles) continuously transmit packets without knowing the load situation of their relay nodes, network congestion may occur. Besides, some position based routing algorithms [17], [18] unrealistically assume that vehicles are moving randomly on roads without considering the vehicles' regular moving pattern, which is inefficient for packet transmission in VANETs.

To overcome these shortcomings of existing routing protocols, we propose a hierarchical routing scheme with load balancing (HRLB) for SDVN. This mechanism makes full use of the advantages of SDN central controller, and designs a hierarchical routing from both global and local perspectives. Our main contributions can be summarized as followings.

- We consider a novel architecture of SDVN which is of the advantages for achieving centralized route planning and real-time monitoring of the global network load. We focus on the problem of routing with load balancing in SDVN which has not been addressed before.
- Our hierarchical route planning mechanism is put forward respectively from the global and local levels. The first hierarchy is globally as a coarse-grained routing design. The central controller divides the entire urban region space into a number of equal-size grids. A series of grids are selected on the basis of grid vehicle densities and grid transition probabilities. The second hierarchy carries out the fine-grained routing design locally. Firstly, according to path cost function, two transmission paths with low costs value are found out from the region made up of the selected grids. Secondly, on each of these two paths, relay nodes with light traffic loads are chosen relying on a utility function to construct routing.
- Real-world taxis GPS data in Shanghai are exploited to analyze the reasonability of why we adopt the grid density and grid transition probability in grids selection. Also the performance of our proposed mechanism is verified by simulation with the real taxi GPS trajectory data.

The rest of this paper is organized as follows. In Section II, we outline the related work. In Section III, we give the network model and hypothesis of proposed protocol. Our proposed HRLB protocol is described in detail in Section IV. In Section V, simulation results are presented and analyzed. Finally, conclusions are drawn in Section VI.

II. RELATED WORK

In this section, we review some related routing protocols developed for VANETs. Greedy perimeter stateless routing (GPSR) [19] routes packets to the neighboring node with the shortest distance to the destination node. When the current node fails to find a neighbor closer to the destination than itself, the greedy forwarding strategy is switched to a recovery mode. Because GPSR lacks information about the network topology, it can potentially fall into the loops known as the local maximum problem. In [20], a protocol named GPSR-MA-LA is proposed in which each node advertises its load, position, speed and direction of its neighbors to other nodes nearby, and all the intermediate nodes help to decide routing by selecting the best next hop. Although the protocol comprises load balancing mechanism, it only considers the traffic load in a node buffer, and the total traffic load on a road is not taken into account. In [21], geographic load balancing routing protocol in hybrid VANETs, namely GLRV, is designed, which provides a congestion control mechanism for the backbone nodes. However, the protocol implements load balancing mechanism only for Mesh network, but does not execute load balancing for vehicle nodes. Vehicle Density and Load Aware routing protocol for VANETs called VDLA is proposed in [22], which adopts sequential selection of junctions to construct routing. The selection of junctions is depend on the real-time vehicle density, the traffic load of the road segment and the distance to the destination. In VDLA protocol, the collection of the information on vehicle density and traffic load is started by the nodes located at junctions. However in practice, it is hard to guarantee that all intersections have vehicles at any time. VLBR protocol is proposed in [23], which finds the k-Shortest paths between each source and destination according to the related roadąŕs density and length. Packets are forwarded frist on the 1st shortest path. If the collision probability is higher than the predefined threshold, the source vehicle switches to another less congested path to continue sending packets. Nevertheless, the protocol lacks of considering the network load when choosing a good path.

The above schemes shed lights on our design of routing protocol with centralized control. In our scheme, the loads of both the paths and nodes are all taken into consideration, making our approach more effective.

III. NETWORK MODEL AND HYPOTHESIS OF THE PROPOSED PROTOCOL

A. NETWORK MODEL

Consider a SDVN architecture shown in Fig. 1, in which the network is divided into control plane and data plane. The control plane is mainly a central controller managing the whole network operation, and the data plane includes BSs, vehicles and links between BSs. There are two modes of communication implemented in this network, namely V2V communication and V2BSs communication. In V2V communication, the vehicles used to deliver the packets to the intended recipient serve as relay nodes (the terms vehicle and node can be used interchangeably afterwards). Two types of BSs are employed in V2BSs communication, such as macrocell base stations (MBSs) and small base stations (SBSs), both of which adopt LTE technology.

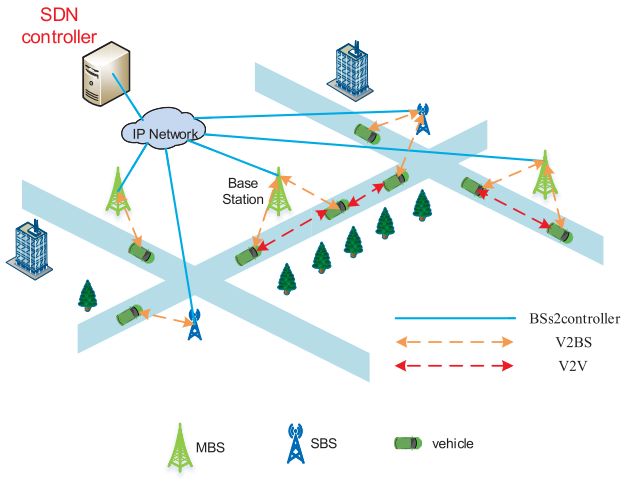


FIGURE 1. SDN-VANETs framework.

All vehicles periodically send beacon messages to the central controller through the BSs for building a global connectivity graph of all vehicle nodes. The beacon message contains position, speed, direction and remaining buffer size.

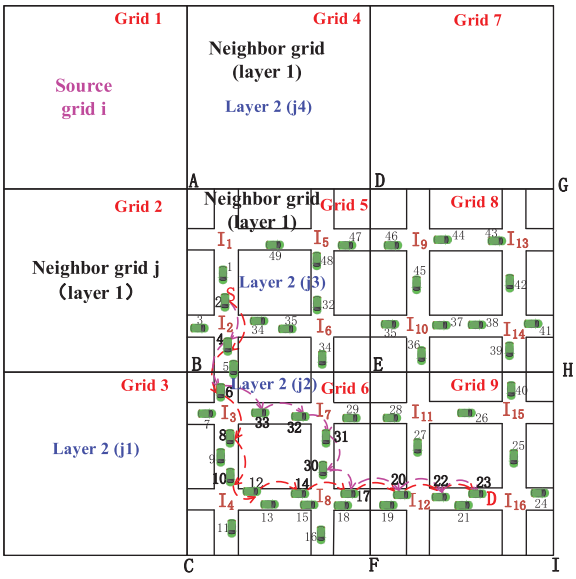


FIGURE 2. A large region is divided into multiple grids.

The central controller divides a large region of a city into grids as illustrated in Fig. 2. The elaborate description of some definitions is as follows:

Definition 1 (Source Grid): It is the grid where the source vehicle is located.

Definition 2 (Destination Grid): It refers to the grid where the destination vehicle is located.

Definition 3 (Grid Vehicle Density): It shows the mean number of vehicles in a specific grid. The grid vehicle density is estimated as $\rho_j = V_j/S_j$, where the V_j denotes number of vehicles in grid j, and the S_j is the area of grid j.

Definition 4 (Grid Transfer Probability): It indicates the probability of a vehicle moves from grid i to grid j, denoted by $p_{i,j}$. The probability is estimated by the ratio of the number of vehicles transferred from grid i to grid j and the total number of vehicles transferred from grid i to all grids.

The reason we consider grid vehicle density and grid transfer probability is as follows. On one hand, for a grid, the higher the density of grid vehicles indicates the better its network connectivity. On the other hand, the size of the grid transfer probability reflects that of the vehicle flow between two grids. The greater the vehicle flow between two grids is, the better their network connectivity obtains.

B. HYPOTHESIS OF PROPOSED PROTOCOL

Our design is on account of the following assumptions which are commonplace in most VANETs geographical routing protocols [14], [24]. All nodes are equipped with satellite based global positioning system (GPS) and navigation systems. The SDN controller and vehicles are installed with a pre-loaded digital map, by which the detailed road topology could be obtained. Besides, the SDN controller maintains a neighboring list of each vehicle in the light of the latest information received from periodically beacon messages sent by vehicles. We assume that the considered urban area has a sufficient number of BSs so that at each moment, each vehicle on the road is covered by at least one BS. Each vehicle is equipped with a LTE radio interface for communication between the vehicle and the BSs. The controller, all BSs and vehicles install the OpenFlow protocol. The controller communicates with network devices via the OpenFlow protocol.

We assume that the transmission rate between the central controller and the vehicles is sufficient high, so the time that the vehicles upload signaling to the central controller and sending routing table to the vehicle nodes is negligible. On the basis of the assumption above, the SDN controller can get the accurate global network statistics.

IV. ALGORITHM DESIGN

A. DESIGN OVERVIEW

Our proposed HRLB scheme takes consideration of the grid vehicle density, grid transfer probability, path length, path vehicle quantity, distance between adjacent vehicles and path network load. In grid selection, we split a large region in urban space into several equal-sized grid regions (referred to as the grids). The major benefit of dividing grids is that after two hierarchies routing planning, that is, coarse-grained and fine-grained, packets can be transmitted through the path with high connectivity in high-density region. The HRLB protocol combines three core algorithms together, that is, grid selection, path selection and relay node selection, as shown in Fig. 3.

- Grid selection: Its purpose is to choose a series of grids with good network connectivity. We define an overall state function as the metric for grid selection. This function consists of grid vehicle density and grid transfer probability.

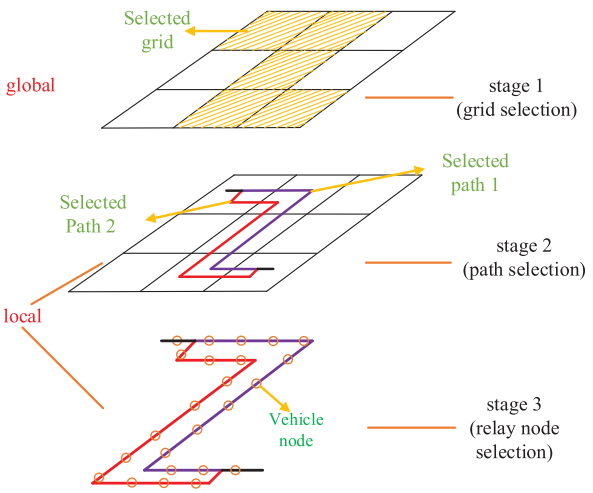


FIGURE 3. Illustration of the proposed hierarchical routing scheme.

- Path selection: A single path consists of several road sections. The goal of path selection is to choose the paths with closer to the destination node, high network connectivity and light load. A path cost function is constructed for path selection. The function is composed of four factors: path length, path vehicle density, adjacent vehicle distance and path network load. The paths in the region consisted of selected grids are sorted based on the size of the path cost function values. At last, two paths with smallest path cost function values are opted.
- Relay node selection: Relay nodes on the selected paths are picked out. Each selected path builds a routing from source to destination. We construct a node utility function for relay nodes selection. This function takes advantage of two factors: the remaining buffer and the distance to the destination.

The HRLB protocol is carried out as follows. When a source vehicle needs to transmit its packets, it searches in advance whether there is a matched routing table. If such a matched routing table exists, the packet is transmitted directly on account of the routing information from this table. Otherwise, a routing request packet will be uploaded by the source vehicle to the central controller. After receiving the request packet, the central controller will first select a train of grids according to the overall state function. Then, in the region made up of selected grids, the central controller sequentially selects two paths with the smallest costs. Finally, applying the node utility function, a series of relay nodes are selected on the two paths. Thus, the central controller obtains two routing tables. We denote the routing table established on the path with the smallest path cost value as the $NO.1$ routing table, while mark another routing table as $NO.$ 2 routing table. The central controller preferentially sends the $NO.1$ routing table to the source vehicle. When the central controller monitors that congestion is about to occur on the path with the minimal cost value, the $NO.2$ routing table will be issued to the source vehicle. After that, the source vehicle switches to new routing table for transmitting the remaining packets.

In the following, the three steps in HRLB are described in details.

B. GRID SELECTION

To select grids with good connectivity through utilizing more grids traffic information, we define a 2-layers neighbor grid below. In Fig. 2, we assume that grid 1 is source grid i with 2-layers of neighbor grids, which includes 3 neighbor grids of layer 1 in total, namely, grid 2, grid 4 and grid 5. Take grid 2 marked as neighbor grid j of the source grid i for an example to explain the meaning of 2-layers neighbor grids. Grid 2 is the layer 1 neighbor grid of the source grid i. While grid 2's neighbor grids become the layer 2 neighbor grids of source grid i (marked as layer 2 (j_m), $m = 1, 2, 3, 4$), except the source grid i itself. The process of grid selection algorithm is shown in Fig. 4. Once receiving routing request message, the central controller immediately checks if source node and destination are in the same grid. If yes, the grid selection terminates; otherwise, continue determining whether their respective grids are adjacent. If they are adjacent, the algorithm terminates; otherwise, the central controller selects appropriate grids relay on overall state function OS. In the process, the central controller prior calculates each neighbor grid j's environment state ρ_ES_j and P_ES_j.

$$\rho\_ES_j = \frac{1}{n}\sum_{k=1}^{n}\rho_{j_k}, \tag{1}$$

$$P\_ES_j = \frac{1}{n}\sum_{k=1}^{n}p^{(2)}_{i,j_k}, \tag{2}$$

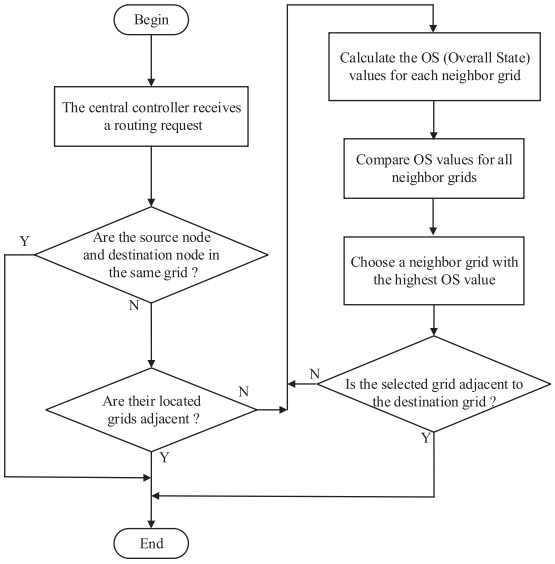


FIGURE 4. Grid selection algorithm.

where ρ_ES_j, P_ES_j and n denote separately the average grid vehicle density, the average grid transfer probability and the

number of all adjacent grids of each neighbor grid j, excluding source grid i. The real-time vehicle density in layer $2(j_k)$ grid is ρ_{j_k}. Furthermore, the 2-order transition probability from the source grid i to grid layer $2(j_k)$ via 2 steps is denoted as $p^{(2)}_{i,j_k}$. When 1-order grid transfer probability matrix has been obtained, $p^{(2)}_{i,j_k}$ can be calculated based on the Chapman-Kolmogorov equation. Thus, the 2-order transfer probability $p^{(2)}_{i,j_k}$ is given by

$$p^{(2)}_{i,j_k} = \sum_{\varsigma \in I} p_{i,\varsigma} p_{\varsigma,j_k}, \tag{3}$$

where I refers to the entire grids. When calculating P_ES_j, we use 2-order grid transfer probability instead of 1-order grid transfer probability. Since the former contains the information of vehicle flow from multiple grids to a certain grid, while the latter only contains the information of vehicle flow from one grid to a certain grid. Therefore, the 2-order transfer probability is greater in terms of geographical range.

Afterwards, the central controller calculates the overall state OS_j of neighbor grid j, defined as

$$OS_j = \mu_1 \frac{\rho\_ES_j}{\rho_{\max}} + \mu_2 \frac{\rho_j}{\rho_{\max}} + \mu_3 p_{i,j} + \mu_4 P\_ES_j, \tag{4}$$

where $\rho_{\max}$ indicates the maximal grid vehicle density in all grid, which is adopted for normalization. We sort all calculated grid densities from large to small and assign the largest grid density value to $\rho_{\max}$. μ_1, μ_2, μ_3, and μ_4 are weighting factors.

After calculating the OS of all neighborhood grids on source grid i, the central controller compares them and filters out the neighbor grid with the maximum OS value. The algorithm terminates if the destination grid is adjacent to the selected grid; otherwise, the above method continues to select next grid.

To show the rationality of considering grid vehicle density and grid transfer probability in grids selection, we conduct big data analysis in **Appendix A** and **Appendix B**. The number of grid vehicles, and the grid transfer probability are analyzed, respectively.

From the results, we conclude that the number of vehicles in a grid is greatly discrepancy and the ratio of vehicles moving from one grid to others varies largely. Moreover, in a relatively short period, vehicles mainly stay in their own grids or move to their adjacent grids. Therefore, it makes sense to apply the two factors for our routing design.

C. PATH SELECTION WITH LOAD BALANCING

Two paths with the smallest value of path cost function will be found out in the region composed of selected grids. Since the central controller has a global view, the path comparison method can be used when selecting paths. Each path contains multiple road sections. Suppose the undirected graph shown as the Fig. 5 is abstracted from urban road network. In this graph, vertex I_i $(i = 1, 2, 3, 4)$ indicates intersection and edge

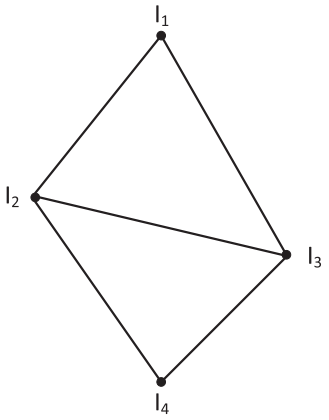


FIGURE 5. Example of the path graph.

$(I_i,\ I_j)$ refers to road segment. From I_1 to I_4, there are 4 paths in total, namely $I_1 \to I_2 \to I_4$, $I_1 \to I_2 \to I_3 \to I_4$, $I_1 \to I_3 \to I_2 \to I_4$ and $I_1 \to I_3 \to I_4$, respectively.

In the following we will detail the path selection algorithm with load balancing. Suppose that there are N paths from source vehicle position to destination vehicle location in the selected grids, which can be represented as a sequence $L = \{l_1, l_2, \cdots, l_N\}$, where $l_i \in L$, $1 \le i \le N$ is the length of the path i. We construct the path cost function PC for selecting path, given by

$$PC_i = \partial \times f(l_i) + \beta \times [1 - g(v_i)] + \gamma \times h(\sigma_i) - \zeta \times \left[1 - \varphi\left(L^i_{avg}\right)\right], \tag{5}$$

in which ∂, β, γ and ζ are weighting factors. The PC_i is determined by the path length function $f(l_i)$, the number of vehicles function $g(v_i)$, the distance of adjacent vehicles function $h(\sigma_i)$, and the path load function $\varphi\left(L^i_{avg}\right)$.

The path length function is

$$f(l_i) = \frac{l_i}{l_{\max}}, \tag{6}$$

where l_i is the curve metric distance from the current intersection to the candidate intersection closest to the destination node. This curve metric distance shows the distance measured when following the geometric shape of a path. $l_{\max}$ is the longest path among N paths.

The number of vehicles is

$$g(v_i) = \frac{v_i}{v_{\max}}, \tag{7}$$

where v_i represents the number of vehicles on the path i. The central controller can count the number of vehicles on the path i on account of the coordinates of each vehicle. $v_{\max}$ denotes the maximum number of vehicles owned by some path among N paths.

The distance of adjacent vehicles is expressed as

$$h(\sigma_i) = \frac{\sigma_i}{\sigma_{\max}}, \tag{8}$$

where $\sigma_{\max}$ indicates the maximum standard deviation among N paths. σ_i refers to the standard deviation of the distance between adjacent vehicles on the path i, given by

$$\sigma_i = \sqrt{\frac{\left(r^i_{a,b} - \mu_i\right)^2 + \left(r^i_{b,c} - \mu_i\right)^2 + \cdots + \left(r^i_{n,m} - \mu_i\right)^2}{N^i_{inval}}}, \tag{9}$$

where $r^i_{n,m}$ denotes interval distance between two adjacent vehicles n and m, and is expressed as $r^i_{n,m} = \sqrt{(x_n - x_m)^2 + (y_n - y_m)^2}$ in which x and y are the longitude and latitude of a vehicle. Moreover μ_i indicates the arithmetic mean of the interval distance between adjacent vehicles on the path i, calculated as

$$\mu_i = \frac{r^i_{a,b} + r^i_{b,c} + \cdots + r^i_{n,m}}{N^i_{inval}}, \tag{10}$$

where N^i_{inval} denotes the total number of intervals on the path i. The reason we take the distance between adjacent vehicles into account is that vehicle distribution is a key factor influencing network connectivity. The more dense the vehicle is, the better the connectivity will be.

The path load is formulated as

$$\varphi\left(L^i_{avg}\right) = \frac{L^i_{avg}}{R_{tr}}, \tag{11}$$

where L^i_{avg} is the average buffer queue length of each vehicle on path i, and $L^i_{avg} = L^i_{sum}/N^i_{sum}$, where L^i_{sum} and N^i_{sum} are the sum of all vehicle's buffer queue length and total number of vehicles on path i, respectively. The R_{tr} is the channel bandwidth.

The central controller calculates eq. (5) for N paths in turn, and chooses two paths with the smallest value of eq. (5).

D. RELAY NODES SELECTION

The central controller selects relay nodes on each selected path. The utility function U is constructed to select relay nodes. The utility function of node i, denoted as U_i, is given as

$$U_i = \frac{B_i}{S_i}, \tag{12}$$

where B_i is the size of remaining buffer in vehicle i and S_i is the distance of vehicle i to destination.

The central controller selects a node with the smallest value of U among neighbor nodes as the relay node. In order to prevent the ping-pong effect during transmission, the central controller only checks those neighbor nodes which are geographically closer to the next intersection or destination than source or relay node. Assuming that the vehicle A has no neighbor nodes as shown in Fig. 6, the central controller will directly select vehicle B as next hop of the vehicle A. Afterwards, the selection of the remaining relay nodes continues to be performed on account of the above method. When a selected node is the neighbor of the destination node, the relay selection process is completed. Finally, we get a transmission routing on each of the two paths. The central controller forms these two routings into their respective routing tables. The two routing tables are marked as *NO*. 1 and *NO*. 2 routing table, respectively. The *PC* value of the path where the *NO*.1 routing table located is smaller than that of the path where the *NO*.2 routing table located. The *NO*. 2 routing table is acted as a backup routing.

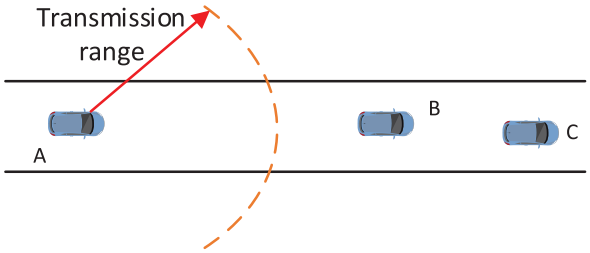


FIGURE 6. Description of no neighbor nodes.

The central controller preferentially issues the *NO*.1 routing table to the corresponding vehicles. Packets sent by the source node are transmitted depending on this routing table. When a relay node cannot find its next hop, the carrying and forwarding mechanism is adopted. While packets are being transmitted, the central controller real-time monitors the load status of the transmission path. Once it is monitored that the sum of the loads of all vehicles on the path selected as *NO*.1 routing exceeds 70% of their total buffer space, congestion on this path is about to occur. At this point, the central controller downloads the *NO*.2 routing table to the homologous vehicles on the path with the second smallest *PC* value. Then, the source vehicle switches to the new routing table to send packets. This approach can prevent packet loss caused by congestion, and reduce the load on the previous more attractive path. In the process of packets transmission, CSMA/CA protocol is used to decrease the probability of collision.

V. AN EXAMPLE OF HRLB PROTOCOL

An example for explaining the proposed protocol is given as follows. Assuming the central controller has a digital map (ACIG) as shown in Fig. 2, we consider 4 grids, including ABED (grid 5), BCFE (grid 6), EFIH (grid 9) and DEHG (grid 8).

The source node 2 will send packets to the destination node 23. Before transmission, the node 2 first searches if the existing of matched routing table. In case of the table missing, a routing request packet will be uploaded to the central controller. Upon receiving the request packet, it immediately starts the route planning algorithm. Initially, the central controller determines that node 2 and node 23 are located in grid 5 and grid 9, respectively. Since there are no roads that directly cross grid 5 and grid 9, they are not adjacent grids. So, the central controller first initiates grid selection mechanism

to select corresponding grid. As grid 6 and grid 8 are adjacent to grid 5, a grid with better connectivity are selected based on eq. (4). Assuming that the *OS* value of grid 6 is greater than that of grid 8, then grid 6 is selected. While grid 9 is adjacent to grid 6, grid selection finishes. Next, two paths are chosen on the basis of eq. (5) in the region composed of the grid 5, grid 6 and grid 9. Suppose that path $I_2 \rightarrow I_3 \rightarrow I_4 \rightarrow I_8 \rightarrow I_{12}$ (called path 1) and path $I_2 \rightarrow I_3 \rightarrow I_7 \rightarrow I_8 \rightarrow I_{12}$ (called path 2) are selected, and the *PC* value of path 1 is less than that of path 2. Finally, the relay nodes are chosen according to eq. (12). Suppose that the selected nodes on path 1 are 4, 6, 8, 10, 12, 14, 17, 20 and 22, and the selected nodes on path 2 are 4, 6, 33, 32, 31, 30, 17, 20 and 22. Thus, two routings are planned and can be formed into two routing tables. These routing tables set up on path 1 and path 2 are referred to as *NO*.1 and *NO*.2 routing table, respectively. The *NO*.1 routing table will be preferentially distributed to the relay vehicles on path 1. Afterwards, the packets sent by the node 2 are delivered on path 1 according to *NO*.1 routing table. During packets transmission, the central controller real-time monitors the network load on the path 1. If the load on path 1 is about to reach congestion, the central controller immediately downloads *NO*.2 routing table. After that, packets sent by node 2 are transferred on path 2 in the light of *NO*.2 routing table. In particular, those packets that do not successfully reach the destination node are not retransmitted.

VI. SIMULATION RESULTS

Our proposed HRLB protocol is compared with GPSR, GPSR-MA-LA, RPGR, VDLA, SCGRP through ns-3 simulator, where the GPSR, RPGR and SCGRP are the routing mechanisms without load balancing function. The simulations are conducted with the real GPS vehicular traces of more than 280 taxis collected, at 9:00-9:20 a.m. on May 1st. The chosen an urban region of Shanghai is 3 kilometers in length and 3 kilometers in width. The whole region is divided into 9 equal sized square grids with $1000m \times 1000m$. We randomly select 10 sources and 10 destinations from the 280 taxis for simulation. All packets have the same size and priority. Other parameter settings are listed in Table I. We do not consider retransmissions caused by collision of packets. Nodes perform FIFO queuing method to buffer packets pending for transmission.

As shown in Fig. 7, we compare the performance of these six protocols in terms of delivery ratio. The delivery ratio is defined as the proportion of successfully delivered packets to the total packets to be transmitted. Compared with GPSR, GPSR-MA-LA and VDLA, our mechanism performs better than them in the delivery ratio. GPSR performs worst among these protocols, because it only considers the distance of the candidate nodes to the destination when choosing next hop, which can easily results in the loss of a large number of packets during transmission. GPSR-MA-LA is better than GPSR since the former takes the velocity and load of candidate node into consideration. VDLA has nearly 20% higher delivery ratio than GPSR-MA-LA, which is caused by the reason that

TABLE 1. Simulation parameter.

Parameter Name	Parameter Value
Vehicle Beacon Interval	30 Sec
Number of vehicles	280
Vehicle velocity	$0 \sim 23$ m/s
Number of packets	100∼800
Size of packet	1024 Bytes
Communication range	500 m
Channel bandwidth	6 Mbps
Packet generation interval	10 Sec
MAC protocol	IEEE 802.11 p
Weighted variable: $\mu_1, \mu_2, \mu_3, \mu_4$	0.4, 0.2, 0.2, 0.2
Weighted variable: $\partial, \beta, \gamma, \zeta$	0.8, 0.2, 0.2, 0.5
Size of simulation area	$3000m \times 3000m$
Simulation time	1200 Sec

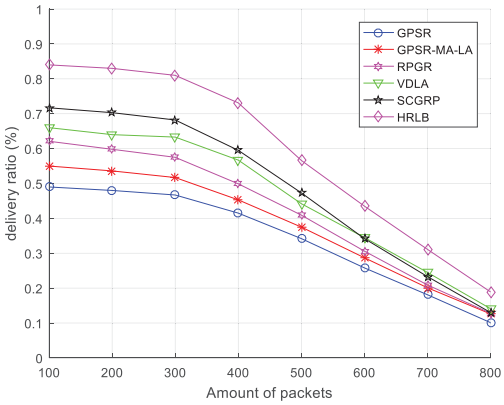


FIGURE 7. Delivery ratio vs. the number of packets.

VDLA selects a well-connected and lightly loaded path to transmit packets. By comparison, the delivery ratio of our mechanism is nearly 28% higher than VDLA. The reason for this is that in our protocol, the path used to transmit packets is the relatively best path on network connectivity and load level. Besides, we choose those nodes with light load as relay nodes. We can also find that when the number of packets increases, the delivery ratio of all algorithms decreases, which is cause by the limited overall capacity of the network. By injecting more packets, the packets can compete for network resources and as a result, fewer packets can be successfully delivered to the destination.

Compared with two other routing mechanisms without load balancing function, RPGR and SCGRP, the proposed HRLB mechanism is nearly 25% and 42% higher than SCGRP and RPGR, respectively. There are two main reasons. On one hand, we select two best transmission path on connectivity in the high density region. On the other hand, the proposed HRLB mechanism could select the path with light loads to transmit packets because of the load balancing function. The delivery ratio of both SCGRP and RPGR are very close to VDLA. This is mainly because both of them are based on road length and connectivity. When the load in the network increases, the delivery ratio of SCGRP and RPGR is

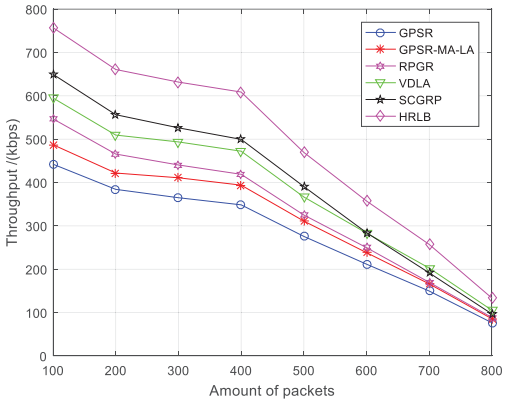


FIGURE 8. Throughput vs. the number of packets.

lighter than VDLA. The truth is that VDLA selects a path with good connectivity and light load to transmit packets, while SCGRP and RPGR do not consider the load factor.

Fig. 8 reflects the throughput of the six protocols. The throughput is defined as the number of bits successfully transmitted per second in the network. In detail, the throughput of GPSR ranges from 76 kbps to 442 kbps; the GPSR-MA-LA is from 85 kbps to 487 kbps; the VDLA increases from 105 kbps to 595 kbps, and the throughput of our mechanism ranges from 133 kbps to 757 kbps. Our protocol can achieve an average gain of about 202, 170 and 105 kbps than GPSR, GPSR-MA-LA, and VDLA protocols, respectively. That is because in the same amount of time, with the implementation of our agreement, more packets can successfully reach their destinations.

For SCGRP and RPGR, the throughput of SCGRP is from 96 kbps to 649kbps, while RPGR ranges from 88kbps to 546kbps. Obviously, the throughput of both is lower than that of the HRLB routing mechanism, because more packets can be delivered to the destination vehicle in HRLB.

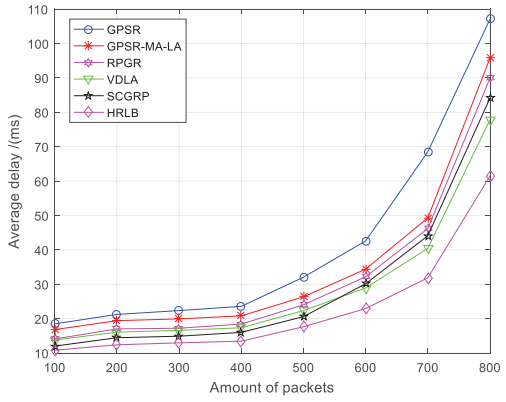


FIGURE 9. Average delay vs. the number of packets.

Fig. 9 illustrates the performance of the average delay. The average delay is defined as the average amount of time experienced by the received packet to reach the destination. Our algorithm has a lower delay than GPSR, GPSR-MA-LA and VDLA. This improvement of our mechanism is primarily because of the truth that high delivery ratio usually result in lower delay.

With the same as other routing protocols, HRLB has less average delay than SCGRP and RPGR, regardless of how many packets are transmitted. This is because that HRLB has higher delivery ratio performance.

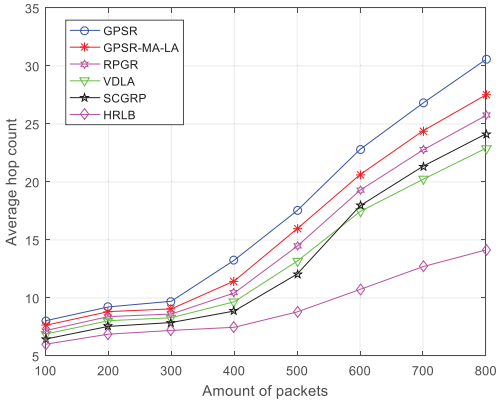


FIGURE 10. Average hop count vs. the number of packets.

Fig. 10 shows the performance of the average hop count. We define the average hop count as the average number of hops that the packets pass through from the source node to the destination node. Apparently, the average hops of our scheme are less than all other three schemes. It can be seen from Fig. 10, the average hop count of all protocols increases with the increase of network load. For GPSR, GPSR-MA-LA and VDLA, with more packets swarm in, each packet waits longer in buffer, so the destination node may have far away from the source node or relay node, thus the packets need go through more hops reaching the destination node. The average hop count increase slightly for the proposed protocol. In our protocol the routing table can fix the number of hops, but when the load increases, switching from the original routing to the backup routing may lead to an increase of hops.

As expected, the hop count performance of SCGRP and RPGR is also inferior to HRLB. The explanation for this reason is the same as GPSR, GPSR-MA-LA and VDLA.

VII. CONCLUSION

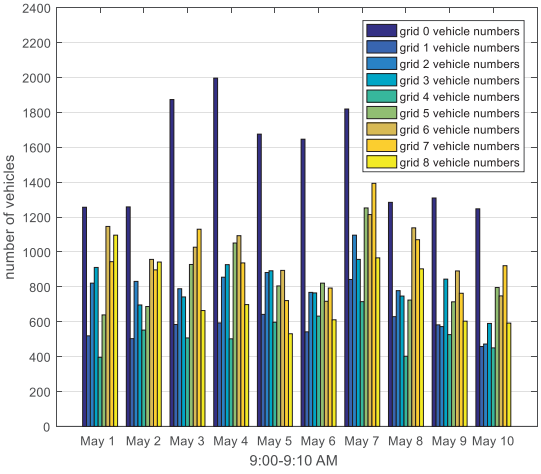
In this paper, we propose a hierarchical routing protocol with load balancing in SDVN to improve packets transmission performance. In our approach, we plan the routing via three stages. In the first stage, a large region space is divided into multiple grids, and those with well-connectivity are selected on account of the grid vehicle density and grid transfer probability. To verify that the grid vehicle density and grid transfer probability considered in the grid selection are reasonable, the big data analyses are conducted with massive datasets of real taxi GPS traces. In the second stage, according to the path cost function constructed by the path length, the number of vehicles, the distance between adjacent vehicles and the

path load, two paths with lowest costs are selected from the region composed of the selected grids. Thirdly, relay nodes are chosen in both of the selected paths depend on the node utility. Finally, we utilize the realistic urban taxis GPS data to verify the performance of our proposed algorithm. Simulation results show that our algorithm outperforms other methods in terms of delivery ratio, throughput, average delay and average hop count.

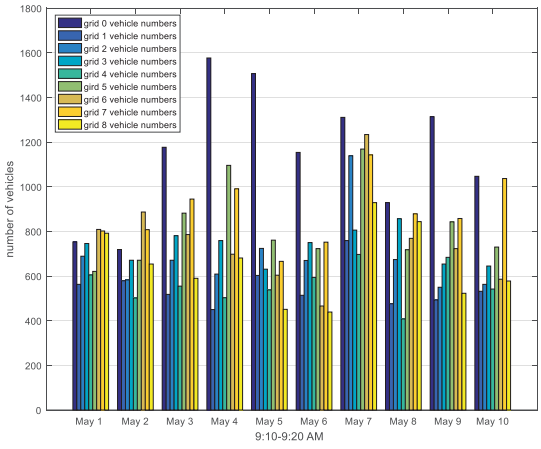
The routing mechanism for urban scenes is the keypoint of this paper. In the future, we plan to study the unicast routing mechanism under expressway scenarios. Expressway is characterized by sparse vehicle density and fast vehicle speeds, which pose greater challenges for reliable routing. In order to ensure that the routing planned by SDN central controller does not lag behind the change of vehicle network topology on expressway, it is necessary for each vehicle to report its coordinates, speed and buffer space timely and accurately. The central controller translates these accurate network statistics into parameters that can be used by machine learning technology and artificial intelligence technology to quickly formulate forwarding routing for the relay vehicles on the expressway.

APPENDIX A
STATISTICS ON THE NUMBER OF GRID VEHICLES

To understand the differences about the number of vehicles between grids, we divide a $3000m \times 3000m$ space in the center of Shanghai, China, into 9 equal square grids labeled as 0-8. The data set used for statistics is a 14-day moving trajectory dataset of 18900 taxis. As can be seen from Fig. 11, the number of taxis between grids is significantly different, which has hundreds of gaps, even more. This fully reveals that there is a great difference in network connectivity between each grid. Intuitively, the vehicle density of a grid characterizes the grid's network connectivity, which is extremely important for routing planning. Hence, the statistical results show that it is reasonable to consider the grid vehicle density when choosing grids.



(a)



(b)

FIGURE 11. Statistics on the number of vehicles in the grids. (a) and (b) are the number of taxis in each grid from 9: 00-9: 10 am, 9: 10-9: 20 am, from May 1 to May 10 in 2016.

APPENDIX B
STATISTICS FOR GRID TRANSFER PROBABILITY

To investigate the characteristics of the vehicle flow between grids, we perform big data analysis by establishing a mathematical model of the grid transfer probability. We first present the spatiotemporal regularity with vehicle mobility, as it inspires us to further count the grid transfer probability.

In the following two-part model, for simplification of discussion, the whole space is divided into Q small square grids, and is denoted by $S = \{s_0, s_1, \cdots, s_{Q-1} | s_i \cap s_j = \emptyset\}$, where Q is the total number of grids.

A. SPATIOTEMPORAL REGULARITY ANALYSIS WITH VEHICLE MOBILITY

The time is slotted in this model. The location of a vehicle at a given time is considered as a random variable with state values of the grid space. Let s^i denote the state variable of vehicle i. We reveal the spatiotemporal regularity by computing the marginal and the conditional entropies of s^i given the previous K states.

For vehicle i, suppose that we have observed its state for N time slots. The state sequence of the vehicle can be denoted by a vector $V_i = \langle s_0^i, s_1^i, \cdots, s_{N-1}^i \rangle$, where $s_n^i \in S$, $0 \leq n \leq N-1$ is the grid state of vehicle i at time slot n. Assuming that state s_n^i appears $m_{s_n^i}$ times in the vector V_i. Then, the probability of vehicle i taking state s_n^i can be computed as $p\left(s_n^i\right) = m_{s_n^i} \Big/ N$. Thus, the marginal entropy of s^i is

$$H\left(s^i\right) = -\sum_{s_n^i=0}^{Q-1} p\left(s_n^i\right) \log_2 p\left(s_n^i\right). \tag{13}$$

Next, we compute the conditional entropy of s^i given its immediately previous state $s^{1,i}$ which has the same

distribution with s^i. The conditional entropy is

$$H\left(s^i|s^{1,i}\right) = H\left(s^i, s^{1,i}\right) - H\left(s^{1,i}\right) = H\left(s^i, s^{1,i}\right) - H\left(s^i\right). \quad (14)$$

To calculate the conditional entropy of eq. (14), we further derive a sequence of 2-tuples, $V_i^1 = \langle (s_0^i, s_1^i), (s_1^i, s_2^i), \cdots, (s_{N-2}^i, s_{N-1}^i)\rangle$. By counting the occurrences of a certain element $\left(s_{n-1}^i, s_n^i\right)$, denoted by $m_{s_{n-1}^i, s_n^i}$, we can get joint probability $p\left(s_{n-1}^i, s_n^i\right) = m_{s_{n-1}^i, s_n^i} \Big/ (N-1)$. Therefore the eq. (14) is rewritten as

$$H\left(s^i|s^{1,i}\right) = -\sum_{s_n^i=0}^{Q-1}\sum_{s_{n-1}^i=0}^{Q-1} \frac{m_{s_{n-1}^i, s_n^i}}{N-1} \log_2\left(\frac{m_{s_{n-1}^i, s_n^i}}{N-1}\right) - H\left(s^i\right). \quad (15)$$

Similarly, we can calculate the conditional entropy of $s^{1,i}$ given its immediately previous K states, $H\left(s^i|s^{1,i}, \cdots, s^{K,i}\right)$.

Essentially, the marginal entropy brings to light the uncertainty of a vehicle reaching a position. The conditional entropy shows the uncertainty of a vehicle moving to a certain location when its fore states are given.

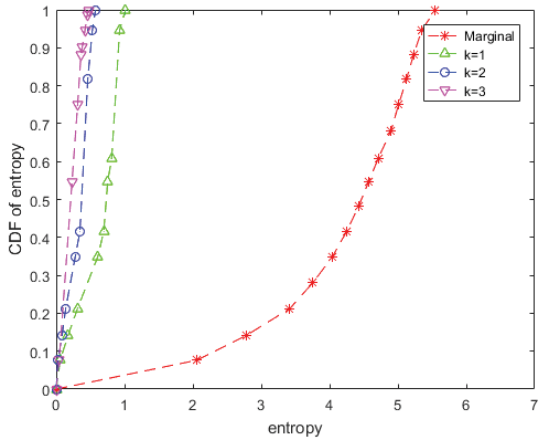


FIGURE 12. The CDFs of marginal entropy and conditional entropies of a vehicle positional state.

Fig. 12 shows the cumulative distribution functions (CDFs) of the marginal and the conditional entropies for $K = 1, 2, 3$. The data set used to draw Fig. 12 is the same as mentioned in Appendix A. We can find the conditional entropies are significantly smaller than the marginal entropy. This hints that the uncertainty of the positional state becomes smaller when the preceding states are known. We also discover that the entropy decreases as K becomes bigger. This indicates that more preceding states help further decrease uncertainty of a vehicle heading for a position.

In the above spatiotemporal regularity, since a vehicle moves from one grid to another with corresponding probability, the number of vehicles transferred from one grid to other each grid will be different. In order to more intuitively investigate the difference of vehicle flow between grids, we explore the grid transfer probability.

B. STATISTICS FOR GRID TRANSFER PROBABILITY

The grid transfer probability reflects the magnitude of the vehicle flow between two grids. The mathematical model of the probability is described in detail below.

A certain period of time T in a day is divided into φ time slots, that is, $\Gamma = \{t_1, \cdots, t_\varphi\}$, where $t_1 = \cdots = t_\varphi$ and t_i denotes the ith time slot. M represents the number of days, $D = \{1, 2, \cdots, M\}$. The time interval of T is the same for these M days.

We define the ith time slot in the jth day as t_i^j and assume that the number of vehicles transferred from grid s_m to grid s_n are $n_{s_m,s_n}^{t_i^j}$ during t_i^j. Then the sequence of vehicle quantity transferred from grid s_m to any one grid during t_i^j can be denoted as $n_{s_m}^{t_i^j} = \left\{n_{s_m,s_0}^{t_i^j}, n_{s_m,s_1}^{t_i^j}, \cdots, n_{s_m,s_{Q-1}}^{t_i^j}\right\}$.

We define A_{T,s_m}^M as the total quantity of vehicles transferred from gird s_m to each grid during T on the Mth day, which is given by

$$A_{T,s_m}^M = \sum_{i\in\Gamma}\sum_{j\in S} n_{s_m,j}^{t_i^M}. \quad (16)$$

Thus, we can get the total vehicle numbers transferred from grid s_m to all grids within M days, written by $H_{s_m}^M$,

$$H_{s_m}^M = \sum_K A_{T,s_m}^K \forall K \in D. \quad (17)$$

Let $B_{s_m,s_n}^{M,T}$ indicate the total number of vehicles from grid s_m to grid s_n during T on Mth day with $B_{s_m,s_n}^{M,T} = \sum_{j\in\Gamma} n_{s_m,s_n}^{t_j^M}$. Then, we can obtain the total number of vehicles transferred from grid s_m to grid s_n within M days, which can be formulated as

$$H_{s_m,s_n}^M = \sum_K B_{s_m,s_n}^{K,T} \forall K \in D. \quad (18)$$

The grid transfer probability from grid s_m to grid s_n is estimated as $P_{s_m,s_n} = H_{s_m,s_n}^M / H_{s_m}^M$. Then we can acquire the 1-order grid transfer probability matrix as

$$P = \begin{bmatrix} P_{s_0,s_0} & P_{s_0,s_1} & \cdots & P_{s_0,s_{Q-1}} \\ P_{s_1,s_0} & P_{s_1,s_1} & \cdots & P_{s_1,s_{Q-1}} \\ \cdots & \cdots & \cdots & \cdots \\ P_{s_{Q-1},s_0} & P_{s_{Q-1},s_1} & \cdots & P_{s_{Q-1},s_{Q-1}} \end{bmatrix}. \quad (19)$$

In order to calculate the grid transfer probability based on above model, we still adopt the same geospatial, grid partition method and data set of vehicle mobility as before. Fig. 13 illustrates the grid transfer probability from grid 0, 1 and 2 to other grids, respectively. As can be seen from these three charts, the number of vehicles moving from one grid to different grids varies dramatically. In a short period of time, vehicles in a certain grid move mainly to their own grids and

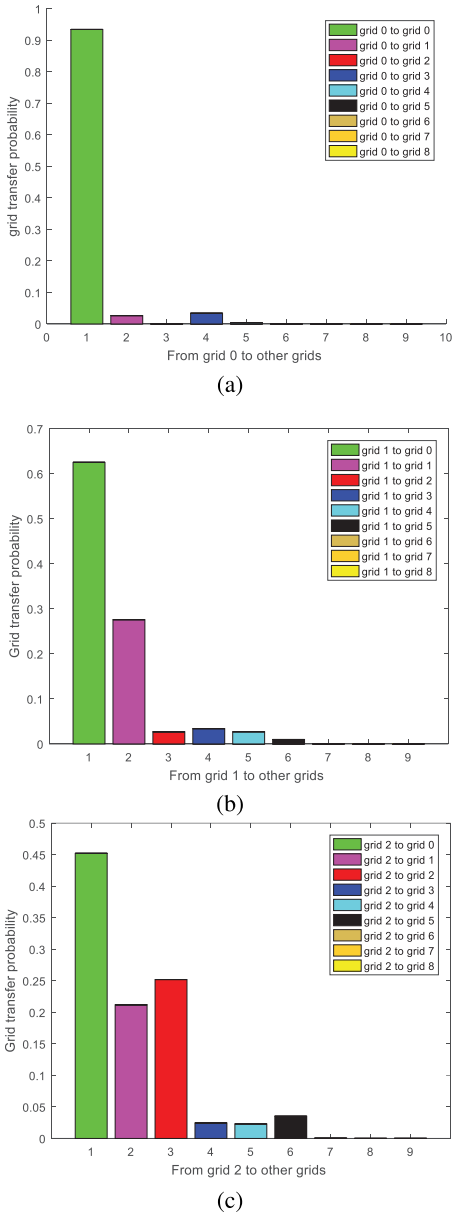


FIGURE 13. Grid transfer probability with vehicle trajectory at 9: 00-9: 10 am. (a), (b) and (c) are the grid transfer probability from grid 0, 1 and 2 to other grids, respectively.

adjacent grids. In addition, the number of vehicles moving to different adjacent grids is also significantly different. Due to these characteristics above, the grid transfer probability becomes an important consideration to select grids with good network connectivity.

REFERENCES

[1] O. S. Oubbati, A. Lakas, F. Zhou, M. Güneş, N. Lagraa, and M. B. Yagoubi, "Intelligent UAV-assisted routing protocol for urban VANETs," *Comput. Commun.*, vol. 107, pp. 93–111, Jul. 2017.

[2] C.-C. Lo and Y.-H. Kuo, "Traffic-aware routing protocol with cooperative coverage-oriented information collection method for VANET," *IET Commun.*, vol. 11, no. 3, pp. 444–450, Feb. 2017.

[3] D. Lin, J. Kang, A. Squicciarini, Y. Wu, S. Gurung, and O. Tonguz, "MoZo: A moving zone based routing protocol using pure V2V communication in VANETs," *IEEE Trans. Mobile Comput.*, vol. 16, no. 5, pp. 1357–1370, May 2017.

[4] S. Kim, "Timed bargaining-based opportunistic routing model for dynamic vehicular ad hoc network," *EURASIP J. Wireless Commun. Netw.*, vol. 2016, Jan. 2016, Art. no. 14.

[5] J. G. Filho, A. Patel, B. L. A. Batista, and J. Celestino, Jr., "A systematic technical survey of DTN and VDTN routing protocols," *Comput. Standards Interfaces*, vol. 48, no. 1, pp. 139–159, Nov. 2016.

[6] P.-J. Chuang and T.-L. Huang, "Efficient vehicular ad-hoc networks routing based on junctions," *IET Commun.*, vol. 9, no. 4, pp. 487–493, 2015.

[7] N. Cordeschi, D. Amendola, M. Shojafar, P. G. V. Naranjo, and E. Baccarelli, "Memory and memoryless optimal time-window controllers for secondary users in vehicular networks," in *Proc. Int. Symp. Perform. Eval. Comput. Telecommun. Syst. (SPECTS)*, 2015, pp. 1–7.

[8] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. 2nd IEEE Workshop Mobile Comput. Syst. Appl. (WMCSA)*, Feb. 1999, pp. 25–26.

[9] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," *Comput. Commun. Rev.*, vol. 24, no. 4, pp. 234–244, Aug. 1994.

[10] H. Saleet, R. Langar, K. Naik, R. Boutaba, A. Nayak, and N. Goel, "Intersection-based geographical routing protocol for VANETs: A proposal and analysis," *IEEE Trans. Veh. Technol.*, vol. 60, no. 9, pp. 4560–4574, Nov. 2011.

[11] N. Alsharif, S. Céspedes, and X. S. Shen, "iCAR: Intersection-based connectivity aware routing in vehicular ad hoc networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2013, pp. 1736–1741.

[12] Y.-S. Chen, Y.-W. Lin, and C.-Y. Pan, "DIR: Diagonal-intersection-based routing protocol for vehicular ad hoc networks," *Telecommun. Syst.*, vol. 46, no. 4, pp. 299–316, Apr. 2011.

[13] C. Chen, Y. Jin, Q. Pei, and N. Zhang, "A connectivity-aware intersection-based routing in VANETs," *EURASIP J. Wireless Commun. Netw.*, vol. 42, Mar. 2014, Art. no. 42.

[14] K. N. Qureshi, A. H. Abdullah, and J. Lloret, "Road perception based geographical routing protocol for vehicular ad hoc networks," *Int. J. Distrib. Sensor Netw.*, vol. 12, no. 2, pp. 1–16, Feb. 2016.

[15] D. K. N. Venkatramana, S. B. Srikantaiah, and J. Moodabidri, "SCGRP: SDN-enabled connectivity-aware geographical routing protocol of VANETs for urban environment," *IET Netw.*, vol. 6, no. 5, pp. 102–111, Sep. 2017.

[16] N. McKeown *et al.*, "OpenFlow: Enabling innovation in campus networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Apr. 2008.

[17] H. Yu, J. Yoo, and S. Ahn, "A VANET routing based on the real-time road vehicle density in the city environment," in *Proc. IEEE 5th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, Jul. 2013, pp. 333–337.

[18] N. Goel, I. Dhyani, and G. Sharma, "An acute position based VANET routing protocol," in *Proc. IEEE Int. Conf. Micro-Electron. Telecommun. Eng. (ICMETE)*, Sep. 2016, pp. 139–144.

[19] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. Int. Conf. Mobile Comput. Netw. (MobiCom)*, 2000, pp. 243–254.

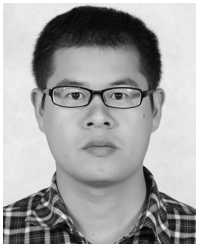
[20] T. D. Nguyen, T. P. Van, T. D. Trong, T. N. Nguyen, H. N. Thi, and H. N. Thi, "A load-balanced and mobility-aware routing protocol for vehicular ad-hoc networks," in *Proc. IEEE Int. Conf. Adv. Technol. Commun. (ATC)*, Aug. 2011, pp. 36–39.

[21] D. Wu, J. Luo, R. Li, and A. Regan, "Geographic load balancing routing in hybrid vehicular ad hoc networks," in *Proc. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2011, pp. 2057–2062.

[22] C. Zhao, C. Li, L. Zhu, H. Lin, and J. Li, "A vehicle density and load aware routing protocol for VANETs in city scenarios," in *Proc. IEEE Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2012, pp. 1–6.

[23] H. T. Hashemi and S. Khorsandi, "Load balanced VANET routing in city environments," in *Proc. IEEE 75th Veh. Technol. Conf. (VTC Spring)*, May 2012, pp. 1–6.

[24] C. Li, C. Zhao, L. Zhu, H. Lin, and J. Li, "Geographic routing protocol for vehicular ad hoc networks in city scenarios: A proposal and analysis," *Int. J. Commun. Syst.*, vol. 27, no. 12, pp. 4126–4143, Jul. 2013.



YANGSHUI GAO received the B.E. degree from the Jiangxi University of Science and Technology, China, in 2009, and the M.S. degree from Yunnan University, China, in 2012. He is currently pursuing the Ph.D. degree with the Beijing University of Posts and Telecommunications, China. He is a member of the Beijing Key Laboratory of Network System Architecture and Convergence. His main research interests include vehicle ad hoc networks and software-defined network.

ZHILONG ZHANG received the B.E. degree in communication engineering from the University of Science and Technology, Beijing, China, in 2007, and the M.S. and Ph.D. degrees in communication and information systems from the Beijing University of Posts and Telecommunications (BUPT), Beijing, in 2010 and 2016, respectively. He is currently a Lecturer at BUPT. From 2010 to 2012, he was a Software Engineer at TD Tech Ltd., Beijing. From 2014 to 2015, he was a Visiting Scholar at Stony Brook University, NY, USA. His research interests include optimization theory and its applications in wireless video transmission, cross-layer design, and wireless networks.

DAN ZHAO received the bachelor's degree from Harbin Engineering University, China, in 2015. She is currently pursuing the master's degree with the Beijing University of Posts and Telecommunications, China. She is a member of the Beijing Key Laboratory of Network System Architecture and Convergence. Her main research interests include the big data analysis and the vehicular ad hoc network.

YI ZHANG received the master's degree from East China Normal University, China, in 2004. He is with the Shanghai Urban-Rural Construction and Transportation Development Research Institute, China. He is the Deputy Director of the Shanghai Transportation Information Center and mainly involved in the research of intelligent transportation system design and traffic data processing technology.

TAO LUO (M'09–SM'15) is currently a Professor at the Beijing University of Posts and Telecommunications and the Beijing Key Laboratory of Network System Architecture and Convergence, China. His research interests include mobile communication, cognitive radio networks, Internet of Vehicles, and machine learning.