INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 15, Issue: 5, Month: October, Year: 2020
Article Number: 3928, https://doi.org/10.15837/ijccc.2020.5.3928

CCC Publications

AGORA
UNIVERSITY PRESS

V2V Routing in VANET Based on Heuristic Q-Learning

X. Y. Yang, W. L. Zhang, H. M. Lu, L. Zhao

Xiaoying Yang
School of Information Engineering, Suzhou University, Suzhou 234000, China
yangxiaoying@ahszu.edu.cn

Wanli Zhang*
School of Information Engineering, Suzhou University, Suzhou 234000, China
*Corresponding author: zhangwnali@ahszu.edu.cn

Hongmei Lu
School of Information Engineering, Suzhou University, Suzhou 234000, China
Hongmeilu@163.com

Liang Zhao
School of Computer Science, Shenyang Aerospace University, Shenyang 110136, China
lzhao@sau.edu.cn

Abstract

Designing efficient routing algorithms in vehicular ad hoc networks (VANETs) plays an important role in the emerging intelligent transportation systems. In this paper, a routing algorithm based on the improved Q-learning is proposed for vehicle-to-vehicle (V2V) communications in VANETs. Firstly, a link maintenance time model is established, and the maintenance time is taken as an important parameter in the design of routing algorithm to ensure the reliability of each hop link. Aiming at the low efficiency and slow convergence of Q-learning, heuristic function and evaluation function are introduced to accelerate the update of Q-value of current optimal action, reduce unnecessary exploration, accelerate the convergence speed of Q-learning process and improve learning efficiency. The learning task is dispersed in each vehicle node in the new routing algorithm and it maintains the reliable routing path by periodically exchanging beacon information with surrounding nodes, guides the node's forwarding action by combining the delay information between nodes to improve the efficiency of data forwarding. The performance of the algorithm is evaluated by NS2 simulator. The results show that the algorithm has a good effect on the package delivery rate and end-to-end delay.

Keywords: V2V communication, VANETs, ITS, Heuristic Q-learning.

1 Introduction

In the past few years, vehicular ad hoc networks (VANETs) have gained a great amount of attention in academia and industry communities [9]. VANET is a key part of Intelligent Transportation System

(ITS) and it includes fixed infrastructures and vehicles, where these vehicles can carry and relay data. Architecture of VANET is shown in Figure 1. If the vehicles in VANET communicate with each other directly, they form a vehicle to vehicle communication (V2V). If the vehicles communicate with a fixed road side unit (RSU), vehicle to infrastructure communication (V2I) will be formed [22] [4]. V2V has the disadvantage of frequent communication disruption caused by the vehicles' joining and leaving the network, while the performance of V2I depends on the communication coverage of RSUs [23]. Designing effective routing algorithms is a challenging task in VANETs, as the applications in ITS, such as driverless technologies and entertainment applications [15] [12], are dependent on vehicular communication.

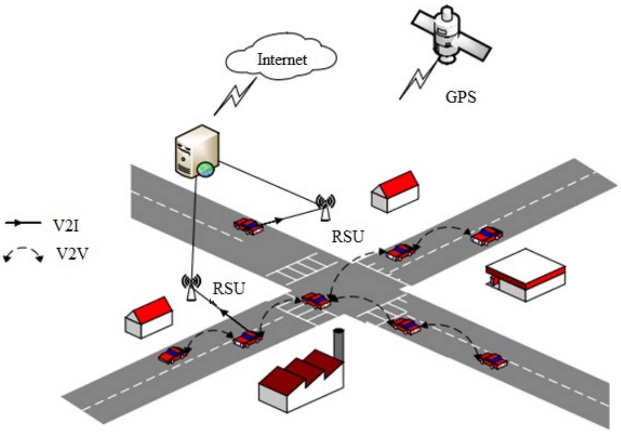


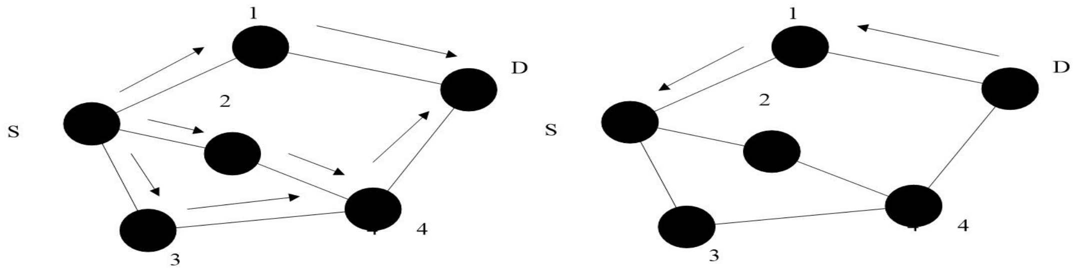
Figure 1: VANET architecture.

Different from the mobile ad-hoc network (MANET), VANET has the characteristics of fast mobile speed of vehicle nodes, short link maintenance time between nodes, frequent disconnection of links leading to extremely unreliable links and complex communication scenarios, which make it difficult for the traditional routing algorithms based on MANET to be applied in the VANET network [17][14][2]. In order to overcome this problem, we need to design a high reliability and high real-time routing algorithm for VANET is needed to design.

Some factors such as the fixed route, the driving direction of two vehicles, distance between vehicles, speed and acceleration can cause the link between the two vehicle nodes to break. As a self-learning algorithm, Q-learning algorithm can find the shortest path from the source node to the destination node in the dynamic environment by constantly interacting with the external information. However, the slow learning convergence of the algorithm results in the problem that the routing algorithm can't quickly reflect the change of topology structure of VANET [11]. Based on this idea, an improved heuristic routing algorithm (HAEQR) is designed based on the improved Q-learning. On the premise of ensuring the reliability of each hop link, HAEQR can accelerate the Q-value update of the current optimal action, improve the learning efficiency and finally accelerate the convergence speed of the Q-learning to adapt to the dynamic network topology of VANET by introducing heuristic function and evaluation function. The contributions that this paper made are as follows. We propose an improved HAEQR based on Q-learning to overcome the slow learning convergence of the routing algorithm that can't quickly reflect the change of topology structure of VANET. On the premise of ensuring the reliability of each hop link, HAEQR can accelerate the Q-value update of the current optimal action, improve the learning efficiency and finally accelerate the convergence speed of the Q learning to adapt to the dynamic network topology of VANET by introducing heuristic function and evaluation function. Simulation evaluation is performed to evaluate the efficacy of the proposed routing algorithm. The rest of the paper is structured as follows. Related work s are reviewed and discussed in section 2. System model is introduced in section 3. The proposed routing algorithm is presented in the section 4. Simulations results are presented in section 5. Section 6 presents the conclusions and the future work.

2 Related work

In VANETs, the routing algorithms are mainly divided into location-based routing algorithm and topology-based routing algorithm [6][16][10][7]. The location-based routing algorithm uses the vehicle's real-time location information to make routing decisions, and determines the packet transmission path according to the location of the destination node and the location of the neighbor node. The typical representative is GPSR [5], which is simple and easy to implement, but it is faced with the problems of routing error and routing interruption because of the fast moving nodes and unstable network topology. In reference [21][8][25] [21], the related improved algorithms are proposed. The routing algorithm based on topology mainly uses the network topology formed by the communication links between nodes to make routing decisions. The typical representative is AODV routing protocol [13]. In AODV, the source node S uses the routing request packet to obtain the route to the destination node D. Each node receiving the routing request packet forwards the routing request packet to its neighbor node, as shown in Figure 2(a), until the routing request packet reaches the destination node D. When the routing request packet arrives at D, D returns the routing reply packet to S. Each node on the return path already contains the routing information to S, then the routing reply packet is forwarded directly according to the routing information, as shown in Figure 2(b).



(a) Broadcast path of Route request packet (b) Broadcast path of route reply packet

Figure 2:

The convergence speed of AODV is delayed as the frequent disconnection of vehicle nodes in VANET resulting in a large number of unreliable routes, which cause the efficiency of routing algorithm to reduce. In reference [18][3][19][1], improved algorithms are proposed. The improved routing algorithm (QLAODV) based on Q-learning method is proposed in reference [18], with grouping as agent, each vehicle node maintains a Q-value table and uses Q-value table as the routing table of node forwarding grouping. QLAODV can sense the change of network topology and the change of communication quality between nodes, dynamically update the Q-value table of nodes. As considering the connectivity between nodes, QLAODV algorithm performs better in delivery package rate and end-to-end delay. However, the Q-value table is updated by the Hello control packet transmitted between vehicle nodes with the problem of slow convergence speed.

In reference [24], the QLAODV algorithm is improved, and a heuristic Q-learning based routing algorithm C-HAQR is proposed. By introducing heuristic function and delay information between nodes to guide the forwarding action of nodes, the learning convergence speed is accelerated. However, the new algorithm does not take into account the link break between the two nodes caused by the vehicle movement, nor does it evaluate the selected actions. It takes more time to explore, resulting in limited learning efficiency.

3 Link reliability model

3.1 System model

In order to effectively evaluate the quality of links between nodes, it is assumed that the road width has little impact on the selection of next hop forwarding nodes, so the road width is ignored, and the expressway is modeled as a case shown in Figure 3.

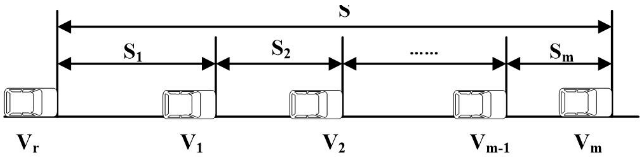


Figure 3: Expressway model.

The vehicles on the road will accelerate, decelerate, change lanes and overtake. The distance between vehicles obeys lognormal distribution, namely $S_i \in logN(\mu_i, \delta_i)$. $S_i = S_i(n), n = 0, 1, 2, ...$ is a random variable from the lognormal distribution (as shown in Figure 3). S_i represents the distance between vehicle i and $i + 1$. $S_i(n)$ is a random variable, representing the distance of node i at time n. In Figure 3, node V_r is taken as the reference node, then the distance from V_r to any node V_n is represented by S, where $S = \sum\limits_{i=1}^{n} S_i$. So S is also subject to lognormal distribution [26].

3.2 Link duration model in one hop range

When the vehicle node drives according to the road model shown in Figure 3, the link between nodes is disconnected mainly caused by two situations: two vehicles running in the same direction and two vehicles running in the opposite direction. Taking vehicle i as the reference node, as sending node, it has the longest link communication time with vehicle j in other cases. The maximum communication radius R of the vehicle is a fixed constant and the maximum speed limit specified on the road is u_max.

At the initial time $t = 0$, the initial speed of any vehicle is $u(0)$ and the initial acceleration is $a(0)$. At any time $t \geq 0$, the acceleration is defined as $a(t)$, and the instantaneous speed is $u(t)$. Acceleration $a(t)$ is calculated according to equation (1):

$$a(t) = \begin{cases} a(0), & t \leq \dfrac{u_{max} - u(0)}{a(0)}, \text{ or } t \leq \dfrac{-u(0)}{a(0)} \\ 0, & \text{else} \end{cases} \tag{1}$$

The speed at time t is calculated according to the speed definition formula (2):

$$u(t) = u(0) + \int_0^t a(T)dT, T \in [0, t] \tag{2}$$

If the vehicle moves at a constant speed, i.e., $a(0) = 0$, then the speed at time t is:

$$u(t) = u(0) \tag{3}$$

If the vehicle moves at an uneven speed i.e., $a(0) \neq 0$, then the speed at time t is:

$$u(t) = \begin{cases} u(t) + a(0)t, & t \leq \dfrac{u_{max} - u(0)}{a(0)}, \text{ or } t \leq \dfrac{-u(0)}{a(0)} \\ u_{max}, & \text{else} \end{cases} \tag{4}$$

The distance travelled by the vehicle with speed of $u(k)$ in time interval [0,t] is defined as:

$$S(t) = \int_0^t u(k)dk, k \in [0, t] \tag{5}$$

Assume that the initial velocity and acceleration of vehicle i and vehicle j are $u_i(0)$, $u_j(0)$, $a_i(0)$ and $a_j(0)$, respectively and the instantaneous acceleration and velocity at time t are $a_i(t)$, $a_j(t)$ and $u_i(t), u_j(t)$, respectively. According to formula (5), the distance between vehicle i and vehicle j in time $[0, t]$ is $S_i(t) = \int_0^t u(k)dk$, $S_j(t) = \int_0^t u(k)dk$, respectively. Vehicle i and vehicle j are in the range of one hop communication, and the initial distance between them is S_0, which satisfies the requirement of $0 \leq S_0 < R$. The distance between vehicle i and vehicle j is D_{ij} at time t. When $D_{ij} > R$, the communication link between vehicle i and vehicle j is disconnected. That is to say, when $D_{ij} = R$, the communication link between vehicle i and vehicle j is in a critical state of disconnection, from which the longest time for maintaining the link between vehicles can be calculated.

3.3 Maximum link maintenance time in the same direction

Vehicle i running in the same direction as vehicle j and vehicle i in front, the distance between vehicles i and j is:

$$D_{ij} = S_i(t) - S_j(t) - S_0 \tag{6}$$

where $D_{ij} = R$, which is satisfied with the condition of equation (7), the maximum link maintenance time between vehicles can be calculated:

$$S_j(t) - S_i(t) + S_0 = -R \tag{7}$$

Due to $S_j(t) - S_i(t) = 1/2a_nt^2 + u_nt$, maximum link maintenance time between vehicles i and j is:

$$t_{i,j} = \frac{-u_n - \sqrt{u_n^2 - 2a_n(R + S_0)}}{a_n} \tag{8}$$

where, $a_n = a_j - a_i, u_n = u_j - u_i$.

Vehicle i running in the same direction as vehicle j and vehicle j in front, the distance between vehicles i and j is:

$$D_{ij} = S_j(t) - S_i(t) + S_0 \tag{9}$$

when $D_{ij} = R$, which is satisfied with the condition of equation (10), the maximum link maintenance time between vehicles can be calculated:

$$S_j(t) - S_i(t) + S_0 = R \tag{10}$$

The maximum link maintenance time between vehicles i and j is:

$$t_{i,j} = \frac{-u_n - \sqrt{u_n^2 - 2a_n(S_0 - R)}}{a_n} \tag{11}$$

where, $a_n = a_j - a_i, u_n = u_j - u_i$.

3.4 Maximum link maintenance time in opposite direction

Vehicles i and j driving in opposite directions, the distance between them is:

$$D_ij = S_i(t) + S_j(t) + S_0 \tag{12}$$

where, $D_{ij} = R$, which is satisfied with the condition of equation (13), the maximum link maintenance time between vehicles can be calculated:

$$S_j(t) + S_i(t) + S_0 = R \tag{13}$$

Due to $S_j(t) + S_i(t) = 1/2a_nt^2 + u_nt$, maximum link maintenance time between vehicles i and j is:

$$t_{i,j} = \frac{-u_n + \sqrt{u_n^2 - 2a_n(S_0 - R)}}{a_n} \tag{14}$$

where, $a_n = a_i + a_j, u_n = u_i + u_j$.

4 The HAEQR routing algorithm

4.1 The QLAODV routing algorithm

A distributed routing algorithm QLAODV based on Q-learning is proposed by Celimuge WU et al. in 2010. The Q-learning process is shown in Figure 4. In a state s, select an action a to execute and move to the next state. The agent evaluates the "state action" value (i.e. Q-value) according to the reward value of environment feedback and the next state after the action is executed. When

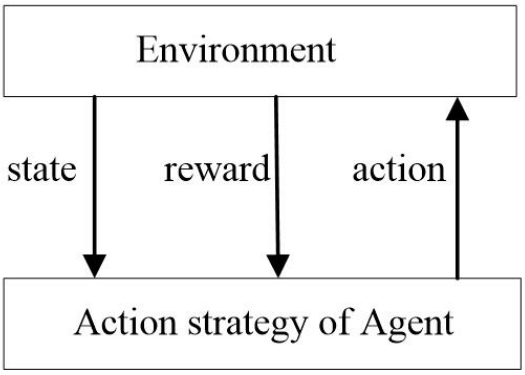


Figure 4: Q-learning process.

Table 1: Table of Q-value.

Neighbor node	G			
	d_1	d_2	...	d_n
x_1	$Q(d_1, x_1)$	$Q(d_2, x_1)$	...	$Q(d_n, x_1)$
x_2	$Q(d_1, x_2)$	$Q(d_2, x_2)$	...	$Q(d_n, x_2)$
...	...	...	...	...
x_n	$Q(d_1, x_n)$	$Q(d_2, x_n)$	...	$Q(d_n, x_n)$

the agent tries every action in each state repeatedly, it learns the optimal strategy in each state by continuously interacting with the environment, obtaining reward value and updating Q-value.

Several definitions of QLAODV routing algorithm are given below.

Define 1: basic components Learning environment: take the whole VANET as the learning environment of agent. Agent: Each data packet $P(o, d)$ can be regarded as an agent. State space: all vehicle node in VANET are packet state spaces. Action: Action sets of the node state are all one hop neighbor nodes. When a node forwards a packet to its next hop neighbor node indicates the state of the packet changes. Immediate reward R: the agent's immediate reward for an activity.

Definition 2: Reward value The value obtained by an activity of the agent is called reward value, i.e. Q-value, which ranges from[0,1]. Because the neighbor node jumping from the destination node can reach the destination node directly, the reward value is set to 1. The initial reward value in the whole network is defined as fixed value R by equation (15):

$$R = \begin{cases} 1, & \text{if } s \in N_d \\ 0, & \text{otherwise} \end{cases} \tag{15}$$

where, N_d represents set of a hop neighbor node of destination node d.

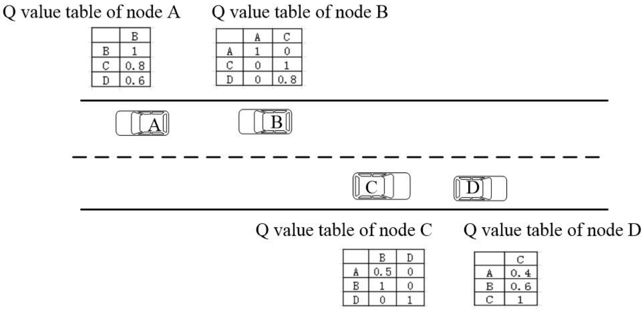
Definition 3: table of Q-value Each node maintains a table of Q-value. When the node selects the next hop node for packet, it directly selects the next hop node with the highest Q-value. The table of Q-value is shown in Table 1, where $Q(d, x)$ indicates that the packet is currently in node s and x is selected as the next hop node arriving at the destination node d with Q-value.

The Q-value is updated by periodically exchanging packets between nodes. The task of learning is distributed to every node, so that the algorithm can quickly converge to the optimal path and make timely adjustments to the changes of network topology. In the QLAODV algorithm, each node updates the Q-value through the Hello package. The updated formula is shown in (16):

$$Q_s(d, x) \leftarrow (1 - \alpha)Q_s(d, x) + \alpha\{R + \gamma \underset{y \in \tau(x)}{max} Q_s(d, y)\} \tag{16}$$

where, $Q_s(d, x)$ expresses the Q-value of that node s selects neighbor node x as the next hop node to forward the packet to destination node d. α is the learning rate. γ is the discount factor. R is the reward value. $\tau(x)$ represents set of the neighbor node of node x, and $max_{y \in \tau(x)} Q_s(d, y)$ is actually the maximum Q-value from neighbor node x to destination node d.

In QLAODV, each node adds the maximum Q-value to the destination node from its Q table and the corresponding neighbor node to the Hello package, and the node receiving the Hello package can use Eq. (16) to update the Q-value table once. When node s needs to send or forward a packet to node d, it only needs to check its own Q-value table, and selects the x with the largest $Q_s(d, x)$ as the next hop node, as shown in Figure 5. If B needs to forward the packet to the destination node D, B can find out C is the most suitable next hop node by looking up the Q-value table.



Q value table of node A

	B
B	1
C	0. 8
D	0. 6

Q value table of node B

	A	C
A	1	0
C	0	1
D	0	0. 8

Q value table of node C

	B	D
A	0. 5	0
B	1	0
D	0	1

Q value table of node D

	C
A	0. 4
B	0. 6
C	1

Figure 5: Example of Q-value table of node.

4.2 HAEQR algorithm description

The update speed of Q-value of QLAODV is seriously limited by the sending interval of hello packets in the environment of VANET, which directly leads to the problem of slow convergence speed of the algorithm. Based on this, a routing algorithm based on improved heuristic Q-learning method namely HAEQR algorithm is proposed in this paper. In the new routing algorithm, the next hop node is selected through heuristic evaluation strategy and the current optimal action, i.e., the current optimal next node is determined according to the delay feedback information between the source node and the destination node. Heuristic and evaluation function are used to inspire and evaluate the node to update the Q-value of the current optimal action to speed up the convergence speed and improve the efficiency of the routing algorithm.

4.3 Update Q-value of HAEQR algorithm

In VANET, the main factors that affect the performance of routing algorithm include vehicle mobility and available bandwidth. In view of the mobility of vehicles between nodes, the delivery rate of packets can be improved by using the link duration model in one hop range.

The first term on the right of Q-value update formula (16) equals the original value of Q-value, and the second term is the latest learning value of Q-value. The latest learning value is generally multiplied by a weight factor, i.e., learning rate. The higher the learning rate is, the greater the proportion of the latest learning value in the updated Q-value is, the faster the learning speed is. In HAEQR, the ratio of link duration between nodes is used as the learning rate of Q-value updating formula. The longer the link duration between nodes is, the faster the learning speed is. To a certain extent, it reduces the impact on packet delivery caused by vehicle mobility and vehicle link instability and improves the delivery rate of packets. The ratio of link duration between nodes is calculated by equation (17):

$$DT_x = \frac{t_{xi}}{\sum t} \tag{17}$$

where, t_{xi} is the link duration between vehicle nodes x and i. $\sum t$ is the sum of the link duration between the current node x and all nodes in the range of one hop.

Discount factor is an important parameter which affects the reward value of an activity of this node according to formula (16). In the VANET network, the available bandwidth is an important parameter which determines the packet transmission rate. The bandwidth BW calculation formula can be defined as follows:

$$BW = \frac{8 \times n \times S_B}{T} \tag{18}$$

where, n represents the number of packets sent and received by the node. S_B is the size of the packet, expressed in bytes and T is the time interval. Suppose the maximum bandwidth value of the node BW_{max} is a fixed value. The bandwidth occupied by the sending group is BW_{packeg}. The effective bandwidth is $BW_{avaliable} = BW_{max} - BW_{packeg}$. Effective bandwidth factor represents the bandwidth of the communication link is calculated by equation (19):

$$BF_x = \frac{BW_{avaliable}}{BW_{max}} \tag{19}$$

Take the bandwidth factor as a factor to affect the speed of learning. As the effective bandwidth changing, it determines the learning progress of each vehicle node. The Q-value updating formula (20) of HAEQR algorithm is obtained by modifying formula (16).

$$Q_s(d,x) \leftarrow (1 - DT_x)Q_s(d,x) + DT_x\{R + \gamma \times BF_x \times \underset{y \in \tau(x)}{max} Q_s(d,y)\} \tag{20}$$

Each node uses equation (20) to carry out iterative calculation, the more hops from the destination node, the lower the final reward value of Q-value. Therefore, the final reward value is determined by the three factors of hops, link reliability and bandwidth. By adding the two parameters of bandwidth and link state, the reliable optimal path of nodes from the source to destination can be finally obtained in this dynamic network.

4.4 The improved use explore balance strategy

When selecting the neighbor to forward package, different from QLAODV directly selecting the next hop node with the maximum Q-value to forward package, the action selection rule to modify the standard greedy rule is used in the HAEQR algorithm, as shown in equation (21):

$$a = \begin{cases} arg\, max_{y \in \tau(x)}[Q_s(d,y) + \varepsilon H_s(d,y) + \delta E_s(d,y)], & \text{if } q \leq p \\ a_{random}, & \text{otherwise} \end{cases} \tag{21}$$

where, $Q_s(d,x)$ expresses Q-value of that node s selects neighbor node x as the next hop node to forward the packet to the destination node d. $H_s(d,x)$ expresses the heuristic function for inspiring the current optimal action. $E_s(d,x)$ expresses an evaluation function that evaluates the success rate of the current optimal action. a_{random} expresses randomly selecting a neighbor node to forward packets. ε is a real variable to weigh the influence of heuristic function. δ is a real variable to weigh the impact of the evaluation function. P is proportion for exploration and utilization, which express node s selecting the next hop node with probability p by using Q value, that is, executing utilization strategy, and randomly selecting the next hop node with probability $1 - p$, that is, executing exploration strategy p. The larger the p value is, the smaller the probability of random selection is, with value of 0.9 in this paper. q is a random number of [0,1]. Each node adopts the improved utilization exploration balance strategy and implements the improved utilization strategy or exploration strategy to forward the packet to the next hop node until the packet reaches the destination node d. At the same time, HAEQR records the delay of each node that the packet passes through in each packet, as shown in Figure 6.

In Figure 6, the sending node S sends packets to the destination node D. Due to the influence of the improved use explore balance strategy, packets may follow different paths. For example, there are three paths to D in Figure 6, $SV_{11} \rightarrow V_{12} \rightarrow ... \rightarrow V_{1n} \rightarrow D$, $S \rightarrow V_{21} \rightarrow V_{22} \rightarrow ... \rightarrow V_{2n} \rightarrow D$ and $S \rightarrow V_{31} \rightarrow V_{32} \rightarrow ... \rightarrow V_{3n} \rightarrow D$. At the same time, packets record the delay between nodes of the path that they pass through. HAEQR uses $T(V_m, V_n)$

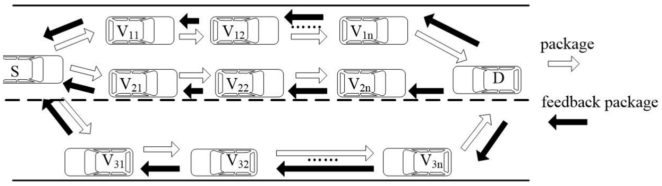


Figure 6: Path of packet and feedback information transmission.

to represent the delay of packet from node V_m to V_n. The delay information of each path between nodes in the figure are $T(S, V_{11}), T(V_{11}, V_{12}), ..., T(V_{1n}, D), T(S, V_{21}), T(V_{21}, V_{22}), ..., T(V_{2n}, D)$ and $T(S, V_{31}), T(V_{31}, V_{32}), ..., T(V_{3n}, D)$.

4.5 Calculate the current optimal next node

The delay between nodes in the path is recorded in the packet. Based on the delay information between nodes, HAEQR can determine the current optimal action of each node in the path. Specifically, every time the destination node receives a packet which has recorded the inter node delay information of all node passing through on the path from the source node to the destination node. If the inter node delay information shows that the time taken for the packet to reach the destination node is shorter than that of the packet in the previous period of time, the destination node immediately takes the inter node delay information of the packet as the feedback information. The feedback information returns to the source node along the path of the corresponding packet. In the process of feedback information return, for each passing node, the node calculates the delay to the destination node according to the feedback information, as shown in equation (22):

$$TD(s, d, x) = \sum_{n=x}^{d-1} T(n, n+1) \tag{22}$$

where, $TD(s, d, x)$ expresses the delay of from node s to select the next hop node x to transmit the packet to the destination node.

Because of using the improved use explore balance strategy, the packet will arrive at the destination node along different paths, and the feedback information will also return to the source node along different paths. For node s, multiple feedback messages may be received from different paths of the same destination node. The feedback message with the shortest delay is the current optimal path from node s to the destination node. The next node corresponding to the path is the current optimal action of packet from node s to the destination node. As shown in equation (23):

$$a_{optimal} = \min_{x \in \tau(x)} TD(s, d, x) \tag{23}$$

where, $a_{optimal}$ expresses the optimal action, i.e. the current optimal next node of node s transmits packets to destination node. $\tau(s)$ is the neighbor set of node s. Through equation (23), the current optimal action of the current node s can be established. Whenever feedback information reaches s, s will recalculate the current optimal action.

4.6 Inspire and evaluate the current optimal next node

When the optimal next node is determined, the current node is inspired to select the current optimal next node by heuristic function $H_s(d, x)$ and the optimal node is evaluated by evaluation function $E_s(d, x)$. The corresponding Q value is updated. The value of heuristics $H_s(d, x)$ affects the choice of action. In order to minimize errors, its value must be as low as possible. It is defined as follows:

$$H_s(d, x) = \begin{cases} max_{y \in \tau(x)} Q_s(d, y) - Q_s(d, x) + \eta, & \text{if } x = a_{optimal} \\ 0, & \text{otherwise} \end{cases} \tag{24}$$

where, $Q_s(d, x)$ expresses the Q-value of s selecting the next hop node x to the destination node d. $max_{y\in\tau(x)}\, Q_s(d, y)$ expresses the max Q-value of one of all neighbor nodes which arrives the destination node d. Equation (25) shows that if neighbor node x is the current optimal next node, it is given an appropriate heuristic value to guide the current node to select the current optimal next node; otherwise, the heuristic value of neighbor node x is 0. η is a very small positive real number, generally 0.01.

For example, in Figure 6, when the sending node S sends packets to the destination node D, the next node can be selected as V_{11}, V_{21}, V_{31}. The Q-values are respectively 0.6,0.5,0.7. The heuristics is V_{11}. If $\eta = 0.01$, $H_s(d, V_{11}) = 0.11$, the H values of other nodes are all 0.

The evaluation function $E_s(d, x)$ evaluates the success rate of the selected node. In the specific learning process, there are many failures in action evaluation. Therefore, this filtering can greatly reduce the number of nodes to be selected and greatly improve the learning efficiency. At the same time, in order to minimize errors, its value must be as low as possible. It is defined as follows:

$$E_s(d, x) = [Q_s(d, y) + \varepsilon H_s(d, y)] - \max_{y\in\tau(x)} [Q_s(d, y) + \varepsilon H_s(d, y)] \tag{25}$$

For example, in Figure 6, when the sending node S sends packets to the destination node D, the next node can be selected as V_{11}, V_{21}, V_{31}. The Q-values are respectively 0.6, 0.5, 0.7. H are respectively 0.11,0,0. If $\varepsilon = 1$, $\rho = 0.01$. The evaluation selects 3 nodes as the next node, and the result is success, failure and failure. Then $E_s(d, V_{11}) = 0.01$, $E_s(d, V_{21}) = -0.2, E_s(d, V_{31}) = 0$.

HAEQR uses heuristic function to guide the node to select the best next node at present, uses evaluation function to evaluate the success rate of the selected node, and accelerates the update of Q value. Specifically, each node adds its own maximum Q-value to the destination node in the feedback information. Whenever a node receives the feedback information, the feedback information contains the maximum Q-value of the previous node reaching the destination node. The node first updates the Q-value of the destination node through the previous node according to equation (21), and then adds the maximum Q-value of the destination node to the feedback information, so that the next node receiving the feedback message updates the Q-value, so as to iterate, until the feedback information returns to the source node, the node on the path completes the Q-value update corresponding to the destination node. With the packet sending and feedback information returning, the Q-value of all nodes in the path from the source node to the destination node will be updated in real time. Therefore, it reduces the influence of too slow Q-value convergence update caused by too long Hello interval, and improves the convergence speed of the routing algorithm.

4.7 HAEQR algorithmic process

HAEQR algorithm mainly updates the Q-value table through Hello packet, and accelerates the update of the current optimal action by using heuristic function and evaluation function, and finally speeds up the convergence speed of the routing algorithm and improves the learning efficiency. The specific steps are as follows:

Initialize settings. Each node initializes Q-value table.

Hello package maintenance process:

Repeat (For each Hello packet sending cycle)

Step 1: Each node adds the maximum Q-value information to all destination nodes in its own table to the packet, broadcasts it to the surrounding neighbor nodes, and transfers to step 2.

Step 2: The current node receives the Hello package from the neighbor node and updates the table of Q-value according to equation (20). Packet transmission process:

Step 1: The current node (including the source node) sends packets to the destination node. Using the improved explore balance strategy (21), select and send the packets to the next node, and turn to step 2.

Step 2: The current node receives the packet, calculates the delay with the previous node, and records it in the packet. If the packet arrives at the destination node, turn to step 3, otherwise turn to step 1.

Step 3: After the destination node receives the packet, if the time delay information between nodes shows that the packet takes less time than the packet in the previous period of time, feedback information will be generated, and the delay information between nodes of the packet and the Q- value to the destination node will be added to the feedback information, and the feedback information will return to the source node along the original path. Turn to step 4.

Step 4: For each node in the return process of feedback information, first, the node uses equation (23) to determine the current optimal action by the delay information of feedback information. Then use equation (24) to calculate the heuristic value of the current optimal action, use equation (25) to calculate the evaluation value of the current optimal action. Then use equation (20) to update its own Q-value table. Finally change the Q-value of feedback information to its maximum Q-value to the destination node and send it to the next node. This is iterated until the feedback information is returned to the source node.

5 Experimental simulation and algorithm performance analysis

In order to evaluate the performance of the algorithm, SUMO is used to generate the node movement scene, and NS2 is used to randomly generate the node data flow. Three routing protocols GPRS, QLOAD and HAEQR are simulated respectively. Basic network parameter configuration is shown in Table 2.

Table 2: NS2 simulation parameter settings.

Parameters	Values
Simulation scene	1800×1800
Vehicle Lane	Two way two lane
MAC Protocol	IEEE 802.11
Transmission range	250m
Simulation time	200S
CBR packet size (byte)	512
Data rate (packet/s)	10
Node velocity (m/S)	$5 \sim 25$
Number of nodes	$50 \sim 250$

5.1 Performance comparison of routing protocols with different number of vehicle nodes

The number of nodes in a certain area will have a great impact on the routing performance of VANET, but also related to the density of network nodes in VANET. The larger the number of vehicles in the same area, the greater the density of nodes is. Keep the maximum speed of nodes at 15m / s.

As shown in Figure 7, the performance comparison diagram of packet delivery rate of three routing protocols under different vehicle nodes is shown. It can be seen that the HAEQR proposed in this paper is significantly better than QLAODV and GPSR. The main reason is that QLAODV updates the Q-value table through the Hello package. The Hello packages are separated too long, it can't reflect the topology change of VANET quickly, resulting in packet loss. HAEQR uses the improved exploration utilization balance strategy, introduces the evaluation function, explores the path with shorter delay and more stable, and determines the current optimal next node. Based on the heuristic Q-learning, HAEQR inspires the node to select the current optimal next node to forward the packet and update the Q-value, which speeds up the Q-value and the convergence speed of learning. Therefore, it can quickly reflect the topology changes, which conducive to the successful delivery of packets to the destination node and improves the communication performance.

As shown in Figure 8, in the scenario of different number of vehicle nodes, the average end-to-end delay comparison of the three algorithms shows that HAEQR is slightly better than GPRS and significantly better than QLAODV. The main reason is that the HAEQR algorithm explores the path with shorter delay and more stable through the improved exploration utilization balance strategy, and the exploration process may take a certain time, but as long as a better path is found, it guides nodes to use the path to transmit packets through the heuristic Q learning heuristic function and evaluation function. It can be seen from the experimental results that the delay benefit of finding a better path through exploration is greater than the time consumed in the exploration process. Therefore, the average end-to-end delay of HAEQR is better than other comparison algorithms.

As shown in Figure 9, the average hops of routes of HAEQR, GPRS and QLAODV are compared. It can be seen that the average hop count of HAEQR is lower than that of GPRS and QLAODV. that is because HAEQR uses feedback information to inspire nodes to select routes with shorter end-to-end delay, which are often routes with fewer hops.

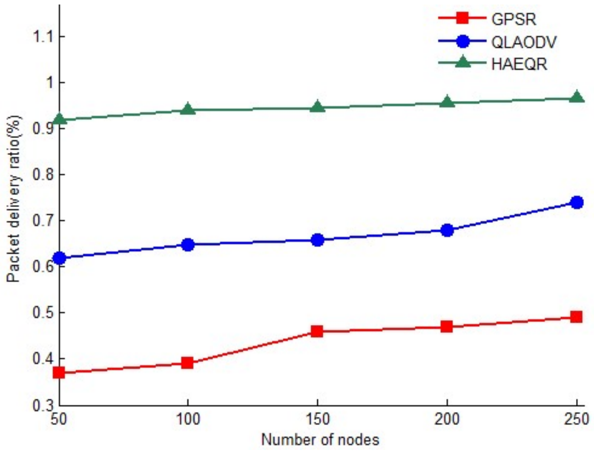


Figure 7: Comparison of packet delivery ratio under different vehicle nodes.

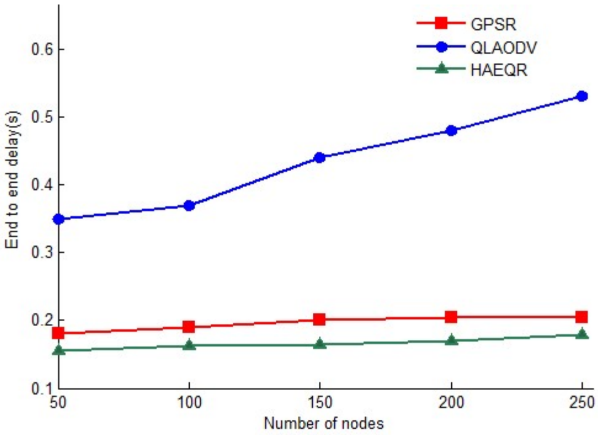


Figure 8: Comparison of end-to-end delay of different vehicle nodes.

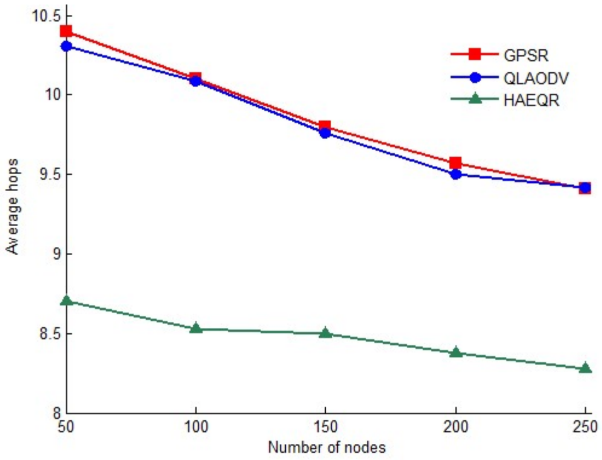


Figure 9: Comparison of Average hops of different vehicle nodes.

5.2 Performance comparison of routing protocols under different vehicle maximum moving speed

The running speed of vehicles in the network has a great impact on the performance of the routing in VANET. The larger the moving speed is, the faster the composition of the network changes, and at the same time, the shorter the communication time is. Keeping the number of vehicle nodes at 100, the performance comparison of packet delivery ratio of three routing protocols is shown in Figure 10.

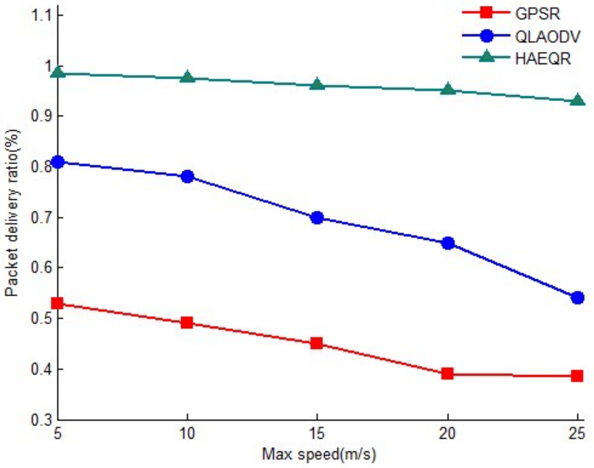


Figure 10: Comparison of Average hops of different vehicle nodes.

As can be seen from Figure 10, with the increase of vehicle speed, the HAEQR proposed in this paper shows a high packet delivery ratio, with an average delivery ratio of more than 90%. However, with the increase of vehicle speed, the other two routing algorithms show a sharp decline in packet delivery ratio. This is because HAEQR fully considers the impact of speed change on link stability. Through the evaluation of the link between nodes, the reliability of the link between nodes is determined. As a learning parameter of HAEQR algorithm, it is applied to the routing decision. GPSR has the lowest delivery ratio, because it does not consider the link reliability when choosing the next hop node and the greedy mechanism is also the main factor to reduce the delivery ratio. Although HAEQR uses Q-learning model, it needs to maintain an end-to-end reliable path, so with the increase of speed, it makes continuous path repair, which leads to the reduction of packet delivery ratio.

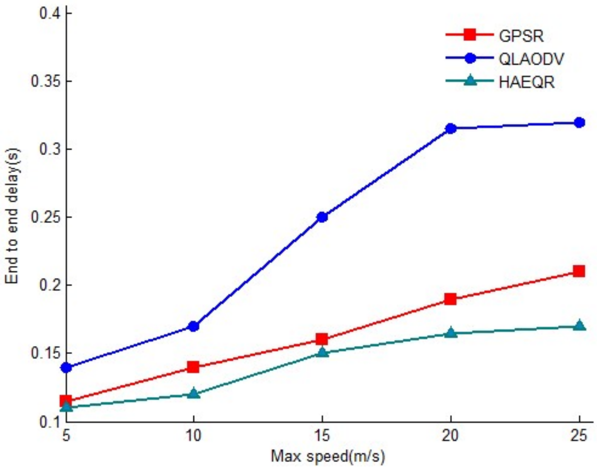


Figure 11: Comparison of end-to-end delay at different moving speeds.

As shown in Figure 11, the end-to-end delay performance of the three routing protocols is compared under different vehicle maximum moving speed. As can be seen from Figure 11, with the increase of vehicle node speed, the delay time of the three algorithms shows an upward trend. It can be seen from the figure that the HAEQR proposed in this paper has the lowest delay and is relatively stable, because the HAEQR algorithm can obtain shorter and more stable path by introducing heuristic function and evaluation function, using shorter exploration delay, adopting link maintenance model to make it less affected by topology, so as to ensure that it has the minimum delay time.

6 Conclusion

In view of the routing problem of the mobile Internet, this paper studies the factors that cause the unreliability of the link by analyzing the characteristics of the vehicle movement and establishes the reliability calculation model of the inter node link. The reliability value of the evaluation of the inter node link is used as a parameter in the improved Q-learning strategy, while the learning convergence in Q-learning is too slow to reflect the VANET quickly. In this paper, heuristic function and evaluation function are introduced to propose an improved routing algorithm based on heuristic Q-learning. Heuristic function is used to speed up learning, evaluation function is used to reduce unnecessary exploration and improve learning efficiency. The experimental results show that the new algorithm has high packet delivery rate and low transmission delay under various conditions, which can effectively solve the problems caused by topology changes.

At present, the success rate of action evaluation function is only divided into success rate and failure rate, which is too rough, however, improving the accuracy of partition will greatly increase the state space affects the learning efficiency. How to find the appropriate partition method is an important problem to be studied in the future study. In addition, due to the limited resources of nodes in VANET, nodes tend to be selfish in order to save their own resources when forwarding packets. If we can consider the selfishness of nodes and dynamically select nodes that actively participate in forwarding, the service quality of VANET will be improved to a certain extent with the help of routing algorithm.

Funding

This work was supported in part by Anhui provincial outstanding young talent support program (gxyq2017093), outstanding academic and technical backbone of Suzhou University (2016XJGG12), the third batch of reserve candidates for academic and technical leaders(2018XJHB07), Suzhou science and technology project (SZ2018GG01, SZ2018GG01xp), New engineering research and practice projects (2017xgkxm54).

Author contributions

The authors contributed equally to this work.

Conflict of interest

The authors declare no conflict of interest.

References

[1] Abbas, N.I.; Ilkan, M.; Ozen, E. (2015). Fuzzy approach to improving route stability of the AODV routing protocol, *EURASIP Journal on Wireless Communications and Networking*, (1), 1-11, 2015.

[2] Agrawal, S.; Tyagi, N.; Iqbal, A.; Rao, R.S. (2020). An intelligent greedy position-based multi-hop routing algorithm for next-hop node selection in VANETs, *Proceedings of the National Academy of Sciences, India Section A: Physical Sciences*, 90(1), 39-47, 2020.

[3] Cao, W.J. (2016). Optimization of AODV protocol for vehicular network, *Nanjing University of Aeronautics and Astronautics*, 2016.

[4] Ghazzai, H.; Ghorbel, M.B.; Kadri, A.; Hossain, M.J.; Menouar, H. (2017). Energy-efficient management of unmanned aerial vehicles for underlay cognitive radio systems, *IEEE Transactions on Green Communications and Networking*, 1(4), 434-443, 2017.

[5] Karp, B.; Kung, H.T. (2000). GPSR: Greedy perimeter stateless routing for wireless networks, *In Proceedings of the 6th annual international conference on Mobile computing and networking*, 243-254, 2020.

[6] Kim, S. (2019). Effective crowdsensing and routing algorithms for next generation vehicular networks, *Wireless Networks*, 25(4), 1815-1827, 2019.

[7] Kumar, I.; Sachan, V.; Shankar, R.; Mishra, R.K. (2018). An investigation of wireless S-DF hybrid satellite terrestrial relaying network over time selective fading channel, *Traitement Du Signal*, 35(2), 103-120, 2018.

[8] Li, C.; Han, J.H.; Wei, Z.C. (2015). GPSR-R routing algorithm in VANET Scenario, *Journal of Hefei University of Technology*, 38(2), 181-185, 2015.

[9] Li, F.; Song, X.; Chen, H.; Li, X.; Wang, Y. (2018). Hierarchical routing for vehicular ad hoc networks via reinforcement learning, *IEEE Transactions on Vehicular Technology*, 68(2), 1852-1865, 2018.

[10] Li, Y.; Shi, D.; Bu, F. (2019). Automatic recognition of rock images based on convolutional neural network and discrete cosine transform, *Traitement du Signal*, 36(5), 463-469, 2019.

[11] Li, G.; Sun, Q.; Boukhatem, L., Wu, J.S.; Yang, J. (2019). Intelligent vehicle-to-vehicle charging navigation for mobile electric vehicles via VANET-based communication, *IEEE Access*, 7, 170888-170906, 2019.

[12] Lin, D.; Kang, J.; Squicciarini, A.; Wu, Y.; Gurung, S.; Tonguz, O. (2016). MoZo: A moving zone based routing protocol using pure V2V communication in VANETs, *IEEE Transactions on Mobile Computing*, 16(5), 1357-1370, 2016.

[13] Perkins, C.; Belding-Royer, E.; Das, S. (2003). RFC3561: Ad hoc on-demand distance vector (AODV) routing, *IEEE Personal Communication*, 36-45, 1997.

[14] Safiulina, A.M.; Ivanets, D.V.; Kudryavtsev, E.M.; Baulin, D.V.; Baulin, V.E.; Tsivadze, A.Y. (2019). Liquid- and solid-phase extraction of uranium(VI), thorium(IV), and rare earth elements(III) from nitric acid solutions using acid-type phosphoryl-containing podands, *Russian Journal of Inorganic Chemistry*, 64(4), 536-542, 2019.

[15] Schoeneich, R.O.; Prus, P. (2018). Improving DTNs performance by reduction of bundles redundancy using clustering algorithm, *International Journal of Computers Communications & Control (IJCCC)*, 13(4), 550-565, 2018.

[16] Senthilkumar, R.; Tamilselvan, G.M.; Kanithan, S.; Arun Vignesh, N. (2019). Routing in WSNs powered by a hybrid energy storage system through a CEAR protocol based on cost welfare and route score metric, *International Journal of Computers Communications & Control (IJCCC)*, 14(2), 233-252, 2019.

[17] Venkatramana, D.K.N.; Srikantaiah S.B.; Moodabidri, J. (2018). CISRP: Connectivity-aware intersection-based shortest path routing protocol for VANETs in urban environments, *Iet Networks*, 7(3), 152-161, 2018.

[18] Wu, C.; Kumekawa, K.; Kato, T. (2010). Distributed reinforcement learning approach for vehicular ad hoc networks, *IEICE transactions on communications*, 93(6), 1431-1442, 2010.

[19] Wu, Z.; Chen, J.; Sun, X.Y.; Qu, L.D. (2017). AODV routing method based on prediction of nodes' moving direction, *Computer Engineering and Design*, 38(9), 2296-2301, 2017.

[20] Xiao, D.G.; Peng, L.X.; Song, D. (2012). Improved GPSR routing algorithm in hybrid VANET environment, *Journal of Software*, 23(S1), 100-107, 2012.

[21] Xiao, D.G.; Peng, L.X.; Song, D. (2012). Improved GPSR routing algorithm in hybrid VANET environment, *Journal of Software*, 23(S1), 100-107, 2012.

[22] Xiao, L.; Lu, X.; Xu, D.; Tang, Y.; Wang, L.; Zhuang, W. (2018). UAV relay in VANETs against smart jamming with reinforcement learning, *IEEE Transactions on Vehicular Technology*, 67(5), 4087-4097, 2018.

[23] Yao, L.; Wang, J.; Wang, X.; Chen, A.L.; Wang, Y.Q. (2017). V2X routing in a VANET based on the hidden Markov model, *IEEE Transactions on Intelligent Transportation Systems*, 1-11, 2017.

[24] Yuan, M. (2017). Research on VANET routing algorithm based on reinforcement learning, *Xi'an: Xi'an University of Electronic Science and technology*, 2017.

[25] Zhang, X.L.; Zhao, Q.; Zhang, T. (2016). Improving GPSR routing protocol in vehicular Ad Hoc network, *Journal of Highway & Transportation Research & Development*, 11(4), 98-103, 2016.

[26] Zhang, D.G.; Ge, H.; Liu, X.H.; Zhang, X.D.; Li, W.B. (2018). A new adaptive routing algorithm based on Q-Learning strategy, *Journal of Electronics*, 46(10), 2325-2332, 2018.

Cite this paper as:
Yang, X. Y.; Zhang, W. L.; Lu, H. M.; Zhao, L. (2020). V2V Routing in VANET Based on Heuristic Q-Learning, *International Journal of Computers Communications & Control*, 15(5), 3928, 2020. https://doi.org/10.15837/ijccc.2020.5.3928