



Novel self-adaptive routing service algorithm for application in VANET

Degan Zhang^{1,2,3} · Ting Zhang^{1,2,3} · Xiaohuan Liu^{1,2,3}

Published online: 14 December 2018
© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract

As a special MANET (mobile ad hoc network), VANET (vehicular ad-hoc network) has two important properties: the network topology changes frequently, and communication links are unreliable. Both properties are caused by vehicle mobility. To predict the reliability of links between vehicles effectively and design a reliable routing service protocol to meet various QoS application requirements, in this paper, details of the motion characteristics of vehicles and the reasons that cause links to go down are analyzed. Then a link duration model based on time duration is proposed. Link reliability is evaluated and used as a key parameter to design a new routing protocol. Quick changes in topology make it a huge challenge to find and maintain the end-to-end optimal path, but the heuristic Q-Learning algorithm can dynamically adjust the routing path through interaction with the surrounding environment. This paper proposes a reliable self-adaptive routing algorithm (RSAR) based on this heuristic service algorithm. By combining the reliability parameter and adjusting the heuristic function, RSAR achieves good performance with VANET. With the NS-2 simulator, RSAR performance is proved. The results show that RSAR is very useful for many VANET applications.

Keywords Vehicular ad-hoc networks · Reliability · Routing service · End-to-end · Self-adaptive · Heuristic algorithm

1 Introduction

With the development of communication technology, the VANET (vehicular ad-hoc network) has become the most important part of building an intelligent city and an intelligent transportation system (ITS). Because it is an extension of traditional MANETs (mobile ad-hoc networks), VANET has elicited much interest from automobile manufacturers, governments, and research institutions to provide services such as traffic warning, information consulting, and entertainment; many

projects in these fields have been launched [1, 2]. According to WHO reports, the number of deaths caused by traffic accidents is 1.24 million worldwide every year. World Bank statistics show that the world economic loss caused by traffic accidents is \$500 billion per year [3]. With greater numbers of vehicles in cities, solving traffic congestion has become a worldwide problem. Traffic congestion can cause time delay, wasted fuel consumption, and unnecessary environmental pollution. At the same time, with the development of a social vehicle network, in-car entertainment and communication between vehicles have become necessary functions that automobile manufacturers provide to customers [4]. In VANETs, wireless communication can be divided into two types: vehicle-to-infrastructure (V2I) and vehicle-to-vehicle (V2V). V2V is favored by many researchers because communication does not rely on infrastructure and its deployment is flexible. However, VANET has some characteristics that are different from traditional MANETs [5]. First, in VANETs, the network nodes (cars) move at high speed, which will cause frequent changes in network topology. The short link time maintained between nodes makes the links unreliable, and network performance varies greatly with vehicle distribution. Second, vehicle nodes are restricted to roads and are affected by many factors like speed limits and traffic signals. Third, vehicles can provide enough power and computing ability for themselves, meaning that the energy limitations of MANET are no longer an important factor.

✉ Xiaohuan Liu
815215568@qq.com

Degan Zhang
gandegande@126.com

Ting Zhang
873432318@qq.com

¹ Key Laboratory of Computer Vision and System (Tianjin University of Technology), Ministry of Education, Tianjin 300384, China

² Tianjin Key Lab of Intelligent Computing & Novel software Technology, Tianjin University of Technology, Tianjin 300384, China

³ School of Electronic and Information Engineering, University of Sydney, Sydney NSW 2006, Australia

As an important support technology to realize an intelligent transportation system, the design of an efficient routing protocol has become an important step in implementing VANET applications. Because the VANET topology changes frequently and the links are unreliable, traditional routing algorithms based on MANETs, such as the Ad-hoc On-demand Distance Vector (AODV) [6] and Dynamic Source Routing (DSR) [7], are difficult to use with VANETs. These algorithms have a lower delivery ratio and a longer transmission delay because they do not consider rapid topology changes. To overcome this problem, some routing algorithms based on geographic location, such as TFOR [8] and GyTAR [9], have been proposed recently [10]. These routing algorithms are designed based on traffic flow estimation, use GPS to locate the destination node, and ignore topology changes, but they do not consider link reliability between nodes. Some reliable routing algorithms have been proposed, such as SLBF [11] and EG-RAODV [12]. These algorithms were designed by adding parameters such as link reliability between nodes and packet error rate. This approach can improve the packet delivery ratio to a certain extent, but cannot guarantee the time delay. In VANETs, a rapid change in velocity is the main cause of topology changes and unreliable links. Only a full understanding of vehicle motion characteristics can lead to effective evaluation of link reliability and efficient routing algorithm designs.

Assuming that vehicles are moving on a fixed route, various factors can break the link between two nodes, such as the traveling direction of the two vehicles, the distance between vehicles, speed, and acceleration. To evaluate link reliability between nodes effectively, these factors should be fully considered, especially given the low-density, high-speed environment. The distance between vehicles plays an important role in ensuring traffic safety and evaluating link reliability. Many models of distance between vehicles have been proposed, such as the exponential distribution model, the normal distribution model, and the lognormal distribution. In [13], the authors proved by experiments that the distance between vehicles can be more consistent with real traffic flow and safe distances when it follows a lognormal distribution. A full understanding of the traffic flow model and the motion characteristics of vehicles can help evaluate link reliability between nodes effectively. With ongoing studies of routing algorithms in recent years, some intelligent algorithms [14] have been applied to VANETs and have yielded better results than traditional routing algorithms. As a self-learning algorithm, the Q-Learning algorithm [15] can find the shortest path from a source node to a destination node through constant interaction with its environment. Based on this idea, in this paper, a reliable adaptive routing algorithm (RSAR) is proposed by improving the Q-Learning algorithm. It can adaptively adjust the Q-table while ensuring the reliability of each hop link to adapt to a dynamic network topology such as VANET.

The main purpose of this paper is to propose a reliable and adaptive routing algorithm. The reliability of an entire link depends on the links between each hop. This paper establishes

a reliable model of a link between nodes by a detailed study of the motion properties of vehicles. Once the probability of link reliability has been calculated, it can be used as a parameter in the Q-Learning algorithm to design the RSAR algorithm.

This paper is organized as follows. In the second section, the routing algorithm is simply classified and partly analyzed. In the third section, the development of a reliable link evaluation model is described. The fourth section presents the main concept of the RSAR algorithm and the development of the model. The fifth section describes the simulation and analysis performed. The sixth and final section provides a summary and suggestions for future work.

2 Related work

2.1 Routing protocol classification

In VANETs, the routing algorithms can be divided into two main types: one based on topology, the other based on geographical location. Routing algorithms based on topology can be divided into reactive (on-demand) routing algorithms and active (table-driven) routing algorithms. In a reactive routing protocol, nodes establish the route according to their requirements and maintain an end-to-end path. Representative routing algorithms include AODV and DSR. Some algorithms like EG-RAODV [12] and QLAODV [15] have been derived on this basis and are more suitable to VANETs. In an active routing algorithm like DSDV and OLSR [16], nodes maintain the topology of the whole network through exchanging the routing table. In a rapidly changing VANET, a fast change in topology is the only factor that affects the active routing algorithm. In a protocol based on geographical location, a GPS is provided for each vehicle so that its coordinates, speed, and direction can be accurately determined. The vehicle does not need to maintain the path from the source to the destination node, and each node maintains only one neighbor table to record the location and speed of neighbor nodes. Routing algorithms based on geographical location can be divided into two kinds. Algorithms in which the sending node makes the routing decision actively are called sending node-based forwarding (SBF). Algorithms in which the receiving node determines whether it can be the next segment of a link are called receiving node-based forwarding (RBF). Among SBF-based routing mechanisms, the most representative is the GPSR [17] routing. In [18], the STAR algorithm is proposed. These are both SBF-based routing algorithms. In RBF-based routing, sending nodes send data packets by broadcasting them. When receiving nodes receive broadcast data packets, they compete for the right to forward them. The nodes that obtain the forwarding right become the next hop in a link and suppress other nodes

from forwarding packets. SLBF [11] is a typical RBF algorithm that uses node locations to calculate waiting time; the nodes that have the shortest waiting time have the right to forward packets. Vegni [19] proposed a clustering-based RBF algorithm that reduces the number of forwarding nodes to suppress redundant packets.

2.2 Link evaluation algorithm

According to routing algorithm classification research, many routing algorithms for VANETs have been proposed, but algorithms that predict link reliability according to motion characteristics of vehicles are rare. In VANETs, it is necessary to ensure link reliability, which guarantees the entire path's validity. In [20], the authors predicted link reliability and routing path according to signal strength. In [21], the authors developed a mobile model to estimate the duration of an n -node path, in which they considered speed, but not acceleration. The MORP [22] algorithm based on motion prediction has also been proposed. MORP predicts the future position of the vehicle and the stability of the link. By using location, direction, and speed to predict link state, MORP selects the most stable end-to-end routing path. A prediction-based routing algorithm [23] has been proposed for VANETs and adjusts the routing path by predicting the life cycle of the link, which is predicted by communication range, vehicle position, and corresponding speed. Considering that the routing path consists of many links, the minimum-duration link determines the maximum survival time of the whole link, so that the maximum link duration of the path can be effectively predicted.

2.3 Q-learning algorithm

The Q-Learning algorithm [24] is a reinforcement learning algorithm, which is proposed in a machine learning context. The learning process is such that in different states, the agent finds a path to reach the destination node with the maximum return value by periodically updating its state-activity value (Q-Value). Q-Learning is an unsupervised active learning algorithm, which does not require a specific system model and can adapt to different environments by interacting with its surroundings. In the Q-Learning algorithm, the Q-value $Q(s, a)$ ($s \in S, a \in A$) represents a reward value that a learning agent converted from state s to another state through an activity a , where s represents a state and a represents an activity. Because a state closer to the destination will receive a higher reward value, the distance between the state and the destination can be effectively determined according to the reward value. As the agent continuously converts activities to find the path to the destination node in different states, it can ultimately find the shortest path from the source node to the destination node. This intelligent algorithm is beginning to be used in VANETs. Some researchers [25–28] have proposed an algorithm by combining fuzzy logic with Q-Learning (FQ-OPP); this algorithm can be adapted to

dynamic topology changes. References [29–32] proposed a QGrid algorithm based on Q-Learning; they divided the entire region into grid cells and selected the next hop forwarding node by calculating the Q-value from both macro and micro aspects.

In addition, recent related studies are described in [33–39]. For example, reference [33] investigated a centralized model and a heuristic algorithm for solving the multi-depot logistics delivery problem, including depot selection and shared commodity delivery. Reference [37] concentrated on introducing an adaptive cooperative caching strategy with novel cache replacement and prefetching policies, which divided the network into non-overlapping clusters. Reference [39] proposed a novel filter model based on a hidden Markov model (HMM) (FM-HMM) for an intrusion detection system to reduce the overhead and time required for detection without impairing the detection rate. Moreover, relative comparisons of results among these methods have been done.

3 Link reliability model

On a highway, the fast movement of vehicles makes links very unreliable. Designing a reliable routing algorithm is a challenging task. The movement of vehicles is the main factor that makes the links unreliable. To evaluate link reliability between nodes effectively, a capable system model and link duration model must be developed. The first step is to develop the system model based on vehicle motion characteristics; then the duration of links between nodes can be evaluated, leading to construction of a reliable link model.

3.1 System model

To evaluate link quality between nodes effectively, assuming that the highway is generally a straight road as in [40, 41] and that the broadcasting distance is much longer than the width of the road, it can be further assumed that the width of the road has a very small influence on selecting the next hop forwarding node, and therefore the width of the road can be ignored. Figure 1 shows the highway model used. It can further be assumed that acceleration, deceleration, changing lanes, overtaking, and other maneuvers are taking place on the road. Besides, the distance between vehicles is log-normally distributed [16], that is, $X_i \in \log N(\mu_i, \delta_i)$. As shown in Fig. 1, $X_i = \{X_i(m), m = 0, 1, 2, \dots\}$ is a random variable that is log-normally distributed. X_i represents the distance between

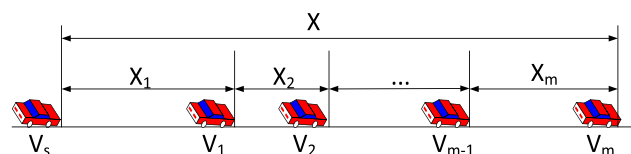


Fig. 1 Road model

vehicles i and $i + 1$, and $X_i(m)$ is a random variable that represents the distance between node i and other nodes at time m . In Fig. 1, if node V_s is regarded as the reference node, then X represents the distance between V_s and any other node, where $X = \sum_{i=1}^m X_i$, so X is also log-normally distributed [42].

Lemma 1 Assuming that $X \in \log N(\mu, \delta)$, the random variable $T = \sqrt{aX + b} + c$ is log-normally distributed, where $a, b, c \in R, a, b, c \neq 0$ and $aX + b \geq 0$.

Proof Let G_T be the probability distribution function of T . For every positive t , $G_T(t) = \Pr[\{T \leq t\}]$. Obviously, because T is continuous,

$$\begin{aligned} G_T(t) &= \Pr[\{T \leq t\}] \\ &= \Pr\left[\left\{\sqrt{aX + b} + c \leq t\right\}\right] \\ &= \Pr\left[\left\{aX \leq (t-c)^2 - b\right\}\right] \\ &= \begin{cases} F_X\left(\frac{(t-c)^2 - b}{a}\right) & \text{when } a > 0 \\ 1 - F_X\left(\frac{(t-c)^2 - b}{a}\right) & \text{when } a < 0 \end{cases} \end{aligned}$$

where F_X is the probability distribution function of X . When $a > 0$, it is clear that T is log-normally distributed. When $a < 0$, according to [13], $F(z) = \frac{1}{2} + \frac{1}{2} \operatorname{erf}\left(\frac{z - \mu_z}{\sigma_z \sqrt{2}}\right)$, and letting $z = ((t - c)^2 - b/a)$,

$$\begin{aligned} 1 - F_X(z) &= \frac{1}{2} - \frac{1}{2} \operatorname{erf}\left(\frac{\ln z - \mu(X)}{\sigma(X) \sqrt{2}}\right) \\ &= F_Y\left(\frac{a}{(t-c)^2 - b}\right) \end{aligned}$$

where Y is a log-normal random variable with parameters $-\mu(X)$ and $\sigma(X)$. Note that the fact that $-\operatorname{erf}(x) = \operatorname{erf}(-x)$ is used. Hence, T obeys a log-normal distribution, completing the proof.

Lemma 2 Assuming that $X \in \log N(\mu, \sigma)$, the random variable $T = aX + b$ is log-normally distributed, where $a, b, c \in R$ and $a, b, c \neq 0$.

This lemma can be easily proved using the arguments aduced in the proof of Lemma 1.

3.2 Link duration model

In Fig. 2, “a” shows two vehicles moving in the same direction, and “b” shows two vehicles moving in opposite directions, where the black arrows indicate the direction of motion.

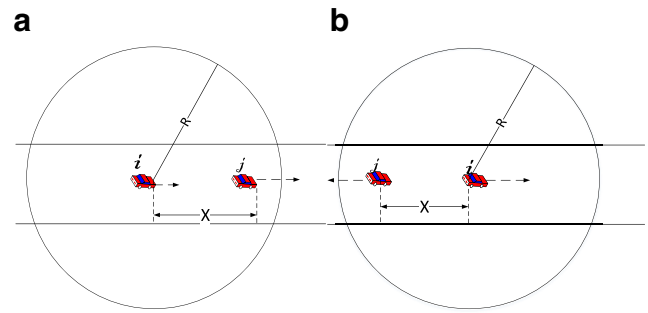


Fig. 2 Two cases of link breakage

Assuming that the vehicles are moving on a fixed road, there are two main conditions (shown in Fig. 2) under which the link between two nodes becomes disconnected. In Figure “a”, two vehicles are moving in the same direction, and in Figure “b”, two vehicles are moving in opposite directions. Consider vehicle i as the reference node and vehicle j as the transmission node. In other cases, vehicles i and j have the longest link communication time. The next step is to analyze the link duration in detail in these two cases.

As shown in Fig. 2a, assume that at time $t_0 = 0$, vehicle i is within the one-hop communication range of vehicle j and that vehicle i is located in front of vehicle j . The initial distance between vehicles X is a random variable. The maximum communication radius of vehicle R is a fixed constant. At the initial moment, X satisfies

$$0 \leq X < R.$$

According to the system model, there will be acceleration, deceleration and overtaking of vehicles on a highway. The maximum speed limit on the road is set to v_m , i.e., all vehicles should move at a speed less than or equal to v_m . Assume that the vehicle’s acceleration is $a(0)$ and that its speed is $v(0)$. When $t \geq 0$, the acceleration is defined as $a(t)$, and the speed is defined as $v(t)$. At time t , the acceleration is as follows:

- 1) If $a(0) = 0$, for all $t \geq 0$,

$$a(t) = 0. \tag{1}$$

- 2) If $a(0) > 0$,

$$a(t) = \begin{cases} a(0) & t \leq \frac{v_m - v(0)}{a(0)} \\ 0 & \text{otherwise} \end{cases}. \tag{2}$$

- 3) If $a(0) < 0$,

$$a(t) = \begin{cases} a(0) & t \leq \frac{-v(0)}{a(0)} \\ 0 & \text{otherwise} \end{cases}. \tag{3}$$

From the above analysis, it can be assumed that when $t_0 = 0$, if the acceleration $a(0)$ at this time is also 0, then the instantaneous acceleration is also 0, i.e., $a(t) = 0$. For Eqs. (2) and (3), if the

speed is below the maximum speed limit or is reduced to 0, it is assumed that the acceleration is still $a(0)$; otherwise, it is changed to 0. Taking the initial speed $v(0)$ into account, at time t , the instantaneous speed $v(t)$ is defined by the following formula:

$$v(t) = v(0) + \int_0^t a(u)du, \tag{4}$$

where $u \in [0, t]$, $a(u)$ is the acceleration at time u .

Now Eqs. (1–4) can be combined to calculate the instantaneous speed:

1) If $a(0) = 0$, for all $t \geq 0$,

$$v(t) = v(0)$$

2) If $a(0) > 0$,

$$v(t) = \begin{cases} v(0) + a(0)t & t \leq \frac{v_m - v(0)}{a(0)} \\ v_m & \text{else} \end{cases} \tag{5}$$

3) If $a(0) < 0$,

$$v(t) = \begin{cases} v(0) + a(0)t & t \leq \frac{-v(0)}{a(0)} \\ v_m & \text{else} \end{cases} \tag{6}$$

According to the above definition, the distance that any vehicle moves at speed $v(x)$ in time interval $[0, t]$ is defined as:

$$S(t) = \int_0^t v(x)dx. \tag{7}$$

Again according to the above definition, as Fig. 2 shows, the distance between vehicles i and j at time t can be calculated. Assuming that the initial speed and acceleration of vehicles i and j are $a_i(0)$, $v_i(0)$, $a_j(0)$, and $v_j(0)$ respectively, the instantaneous speed and acceleration of vehicles i and j at time t are $a_i(t)$, $v_i(t)$, $a_j(t)$, and $v_j(t)$. According to Eqs. (1–7), the distance between vehicles i and j in time interval $[0, t]$ can be calculated as follows:

$$S_i(t) = \int_0^t v_i(x)dx,$$

$$S_j(t) = \int_0^t v_j(x)dx.$$

When the initial distance between vehicles i and j is X , the distance $d_{i,j}$ between i and j at time t is:

$$d_{i,j} = \begin{cases} S_j(t) + S_i(t) + X & \text{same direction} \\ S_j(t) - S_i(t) + X & \text{opposite direction} \end{cases} \tag{8}$$

From Eq. (8), it is obvious that when $d_{i,j} > R$, the link is disconnected.

The next step is to analyze the link duration in Fig. 2b, where two vehicles are moving in opposite directions. When two vehicles meet:

$$S_j(t) + S_i(t) + X = R. \tag{9}$$

The maximum link duration t can now be calculated. By assuming that $S_j(t) + S_i(t) = \frac{1}{2}a_r t^2 + v_r t$, where $a_r = a_i + a_j$ and $v_r = v_i + v_j$, and by substituting this into Eq. (9), the maximum link duration t can be obtained as:

$$t = \frac{-v_r + \sqrt{v_r^2 + 2a_r(R-X)}}{a_r}. \tag{10}$$

When two vehicles are moving in the same direction, as shown in Fig. 2a, during acceleration, deceleration, and overtaking, it is critical to determine whether vehicle i or j is in front. When $S_j(t) - S_i(t) + X > 0$, vehicle j is located in front of vehicle i ; otherwise, vehicle i is located in front of vehicle j . To express effectively which vehicle is in front, a symbolic function was defined as follows:

$$I(i,j) = \begin{cases} 1 & S_j(t) - S_i(t) + X > 0 \\ -1 & \text{otherwise} \end{cases}. \tag{11}$$

When the link is in a critical state of disconnection,:

$$S_j(t) - S_i(t) + X = R \cdot I(i,j). \tag{12}$$

At this point, two cases must be considered to calculate the link duration.

When $I(i,j) = 1$, vehicle j is located in front of vehicle i . From Eq. (12), $S_j(t) - S_i(t) + X = R$. Similarly to Eq. (10), because $S_j(t) - S_i(t) = \frac{1}{2}a_r t^2 + v_r t$, where $a_r = a_j - a_i$ and $v_r = v_j - v_i$, the time t can be determined as:

$$t = \frac{-v_r + \sqrt{v_r^2 + 2a_r(R-X)}}{a_r}. \tag{13}$$

Similarly, when $I(i,j) = -1$, vehicle i is located in front of vehicle j . From Eq. (15), $S_j(t) - S_i(t) + X = -R$, and the link duration t can be determined as:

$$t = \frac{-v_r - \sqrt{v_r^2 - 2a_r(R+X)}}{a_r}. \tag{14}$$

Equations (10), (13), and (14) can be used to calculate the link duration of the sending node and an arbitrary node within one hop range.

Lemma 3 Assuming that the communication link between two vehicles i and j breaks at time t , the link duration time is either a linear function of X or a square root function of X .

Proof When the link disconnects at time t , by the definition of $S_i(t)$ according to Eq. (7), it is known that $S_i(t) = \int_0^t v_i(x)dx$ is a

linear function of t when the speed v_i is constant, i.e., $v_i = v_m$. Let $S_i(t) = at + b$. Similarly, if v_j is a constant, it can be concluded that $S_j(t) = ct + d$ is also a linear function of t . Obviously, according to Eq. (8), when the two vehicles are moving in opposite directions, $(a + c)t + b + d + X = R$. The link duration time $t = \frac{R-b-d-X}{a+c}$ is a linear function. Similarly, when a link has been established between codirectional vehicles i and j , the link duration time is also a linear function according to the equation $(c - a)t - b + d + X = R \cdot I(i, j)$. Hence, the link duration t is a linear function when both v_i and v_j are constant. If either $v_i(t)$ or $v_j(t)$ are not constant, then the distance function will be a quadratic polynomial. Without loss of generality, let $v_i(t) = v_i(0) + a_i t$. By definition, the distance function is:

$$S_i(t) = \int_0^t v_i(0) + a_i \cdot \tau \, d\tau = v_i(0)t + \frac{1}{2} a_i t^2$$

As shown in Eqs. (9) and (12), $S_j(t) \pm S_i(t) = \frac{1}{2} a_r t^2 + v_r t$ and $a \neq 0$. Therefore, the solution must be a square root function of X .

Theorem 1 The duration T of the link between vehicles i and j is log-normally distributed.

Proof By Lemma 3, the link duration can be expressed as either $aX + b$ or $\sqrt{aX + b} + c$. By Lemma 1, the expression $\sqrt{aX + b} + c$ is log-normally distributed. By Lemma 2, $aX + b$ is log-normally distributed. Hence, in all cases, the duration time of the link has a log-normal distribution. This completes the proof of the theorem.

3.3 Link reliability model

Based on the link duration between nodes, the link duration of vehicles i and j is log-normally distributed as $T \in \log N(\mu_t, \sigma_t)$, where the expectation is μ_t and the variance is σ_t . Link reliability can then be evaluated according to duration. Link reliability is defined as the probability of direct communication between two vehicles in time period T_p . Assuming that the communication link between any two vehicle nodes is l , when $t = t_0$, the link reliability between nodes is:

$$r(l) = P\{t_0 + T_p | t_0 \in M\},$$

where M is the starting time for the connection between the two vehicles. Hence, the link reliability is:

$$r_i(l) = \begin{cases} \int_{t_0}^{t_0+T_p} f(T) dT & T_p > 0 \\ 0 & \text{else} \end{cases} \quad (15)$$

where $f(T)$ is the probability function of the time interval T .

4 RSAR algorithm

In VANETs, the fast motion of the nodes makes the network topology change frequently. It is difficult for a simple forwarding strategy to find an efficient routing path in this rapidly changing network environment. Related studies have presented the Q-Learning algorithm as an unsupervised self-learning algorithm. Through continuous interaction with the external environment, it can adaptively adjust its value to find the optimal path to the destination. Hence, it is well suited to dynamic VANETs. This section will describe the routing and forwarding strategy of the Q-Learning algorithm in VANETs.

4.1 Idea of Q-learning algorithm in VANET

Standard Q-Learning is a heuristic learning method based on a learning agent. Generally, in the standard Q-Learning algorithm, the learning process of the agent is composed of a 3-tuple $\{S, A, R\}$, where $S = \{s_1, s_2, \dots, s_n\}$ represents the state space; $A = \{a_1, a_2, \dots, a_n\}$ represents the activity space, and moving from one state to another is regarded as an effective activity; R represents the immediate reward for an activity; and the closer the agent is to the destination, the higher the reward obtained for the activity. The following subsections will present a few related definitions first and then describe the learning process in detail.

4.1.1 Related definitions

Definition 1 Basic components.

Learning environment: The whole VANET environment as the learning environment of the agent;

Agent: Each vehicle node is a learning agent;

State space S : The state space of a given agent is composed of all other nodes except for this agent;

Activity space A : A beacon packet is transmitted from one vehicle to another, which is defined as an activity;

Immediate reward R : obtained when an agent carries out an activity.

Definition 2 Reward value: According to Definition 1, the reward that an agent receives for carrying out an activity is called the immediate reward, and its range is $[0, 1]$. Because the destination node can directly reach the destination node, its reward value is 1. Formula (16) defines the initial value R of the entire network as follows:

$$R = \begin{cases} 1 & s \in N_d \\ 0 & \text{else} \end{cases} \quad (16)$$

where N_d represents the one-hop neighbor node set of destination node D . The reward value of an activity for all the neighbor nodes of the destination node is 1. In the learning

process, the reward value that may be obtained from a transition from one state to another is indicated by the Q-value $Q(s, a)(s \in S, a \in A)$, which lies in the range $[0, 1]$.

Definition 3 Q-table: Each learning agent maintains a two-dimensional table, which is used to record the Q-values of the reachable destination node and its one-hop neighbor nodes. This two-dimensional table is called the Q-table (shown in Table 1).

In the Q-table, the first line contains the IDs of all possible destination nodes, which are expressed as D_i . The first column contains the IDs of the one-hop neighbor nodes, which are expressed as N_i . $Q(D_1, N_1)$ represents the Q-value between the node and the neighbor node N_1 when it reaches the destination node D_1 . As shown in Table 1, the Q-table is a two-dimensional table, its size is determined by the number of neighbor nodes and the number of destination nodes. It is obvious that it has good scalability. The values in the Q-table are updated by periodically exchanging beacon packets between nodes. The learning task is distributed to each node, which makes the algorithm quickly converge to the optimal path, and adjustments to changes in network topology can be made in a timely manner.

4.1.2 Learning process

The above definitions establish that every vehicle node is defined as a learning agent. Unlike traditional ways of learning, each vehicle node has a Q-table. Nodes complete the task of learning by exchanging beacon information and updating their Q-tables. The beacon packet sent by each node contains not only its own speed, location, and other information, but also the maximum Q-value of the neighbor nodes to the destination node, i.e., the maximum value in a column (as shown in Table 2). Assume that there is a VANET topology $G = \{V, E\}$ as shown in Fig. 2, where $V = \{A, B, C, \dots, H\}$ represents the set of vehicle nodes, and that for node A, its state space S_A is the set of all nodes that do not contain A. Edge set E represents a collection of nodes that can communicate directly in one hop range. Assume that as in Fig. 3, A is the source node, and G is the destination node. Now the challenge is to find an optimal path from the sending node A to the destination node G using the self-learning approach.

Learning tasks are assigned to each vehicle node agent, and the learning process is mainly updating the Q-table in the agent and meanwhile updating the state activity pair of the

Table 1 Q-table

	D_1	D_2	...
N_1	$Q(D_1, N_1)$	$Q(D_2, N_1)$...
N_2	$Q(D_1, N_2)$	$Q(D_2, N_2)$...
...

Table 2 Network parameter settings

Parameter	Value
Size of topology (m)	1500×1500
MAC standard	IEEE 802.11 MAC (2 Mbps)
Transmission range (m)	250
Propagation model	Two-ray ground
Simulation time (s)	300
CBR packet size (bytes)	512
Data rate (packets/s)	10

Q-value, $Q(s, a)(s \in S, a \in A)$. The standard Q-Learning function is given as Eq. (17):

$$Q_s(d, x) \leftarrow Q_s(d, x) + \left\{ R + \gamma \cdot \max_{y \in N_x} Q_x(d, y) \right\}, \tag{17}$$

where $Q_s(d, x)$ is the Q-value to be updated, S is the node, x is its neighbor node, N_x is x's neighbor node, D is the destination node, R is the reward value, and $\max_{y \in N_x} Q_x(d, y)$ is the maximum Q-value between x and its neighbor node to the destination node D. The discount factor γ is an important parameter because it affects the reward that a node obtains for carrying out an activity according to Eq. (17). Considering that link stability is an important parameter, the link reliability $r(l)$ between nodes, which was calculated according to Eq. (15), was used as a discount factor, that is, $\gamma = r(l)$. In VANETs, the available bandwidth, as an important parameter, determines the rate of packet transmission. As defined in [28], the bandwidth BW can be calculated as:

$$BW(bps) = \frac{n \times S_B \times 8}{T}, \tag{18}$$

where n represents the number of packets that the node sends and receives, S_B is the size of the packet in bytes, and T is the time interval. Assuming that the maximum available bandwidth of the node is a fixed value defined as \max_{BF} , the bandwidth factor can be calculated as:

$$BF = \frac{\max_{BF} - BW}{\max_{BF}}. \tag{19}$$

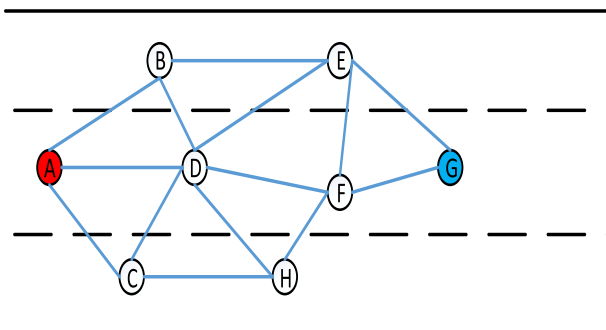


Fig. 3 Model of the learning process

As the factor that affects learning speed, the bandwidth factor varies with changes in effective bandwidth and determines the learning progress of each vehicle node. Equation (17) can be modified to obtain a new heuristic function:

$$Q_s(d, x) \leftarrow (1 - BF) \cdot Q_s(d, x) + BF \cdot \left\{ R + \gamma \cdot \max_{y \in N_x} Q_x(d, y) \right\}. \quad (20)$$

According to Eq. (20), the greater the number of hops, the smaller the reward value. Hence, the final reward value is based on the number of hops, the link reliability, and the bandwidth. By adding bandwidth and link state, the optimal path from source node to destination node can be obtained in the dynamic network. In Fig. 3, nodes E and F are one-hop neighbor nodes of destination node G. According to Eq. (20), the reward values from nodes E and F to destination node G can be represented as $Q_E(G, G)$ and $Q_F(G, G)$, respectively. Considering the effects of link quality and bandwidth, it can be determined that the final Q-values of $Q_E(G, G)$ and $Q_F(G, G)$ are 0.7 and 0.8, respectively. The neighbor nodes of D are A, B, C, E, F, and H. When D receives the beacon packets sent from any neighbor, the data packets are parsed, and the maximum Q-value to the destination node G is extracted, taking $F, \max_{y \in N_F} Q_F(G, Y)$. According to Eq. (20), calculate the corresponding Q-value, i.e., $Q_D(G, F)$, and update the Q-table. Obviously, $Q_F(G, G)$ is the largest, and its value is 1, which will be recorded in the beacon packet. A similar procedure will be performed with data packets from other nodes; then one particular column in the Q-table is updated. Considering bandwidth and link reliability and assuming that $Q_D(G, F) = 0.5$ according to Eq. (20), the Q-tables of the other neighbor nodes are updated, as shown in Fig. 4. As neighbor data packets are continually received, node D constantly modifies the maximum Q-value between node D and its neighbor nodes in the Q-table. Similarly, when node D sends a beacon packet, it traverses a particular column in the Q-table, finds the maximum Q-value among node D and its neighbor nodes, and records and sends the Q-value. When node A receives the

beacon packet sent from node D, it forwards the maximum Q-value and carries on the computation to update $Q_A(G, D)$. The same procedure will be performed when it receives beacon packets from other nodes. Through constant exchange of data packets, the results shown in Fig. 4 will finally be obtained.

According to Fig. 4, the optimal path from A to G is easily found, i.e., the path with the biggest Q-value of its nodes is the optimal path. As Fig. 4 shows, $A \rightarrow B \rightarrow E \rightarrow G$ is the path with the maximum Q-value and therefore is the optimal path. The dynamic update-and-save of the Q-table makes the algorithm respond to dynamic changes in topology quickly and with good reliability as well as robustness.

4.2 Description of the RSAR algorithm

This subsection describes the basic process of data packet forwarding in the RSAR algorithm and then provides a detailed description of each stage.

4.2.1 Basic transmission process

When the RSAR algorithm routes data packets, it performs the following three steps:

1. When a source node sends a data packet, it checks its own Q-table to see whether its first entry is the next hop node to the destination. If so, then it selects the neighbor node with the largest Q-value; if not, then it starts the route development process, which is described in subsection 4.2.2.
2. After the route has been developed, a basic path from the source node to the destination node has been obtained, and the learning of some vehicle nodes is complete. To find the optimal path of the whole network topology and solve the network segmentation problem, the route maintenance process is started to maintain the end-to-end path dynamically. The route maintenance process is described in subsection 4.2.3.
3. Through the processes described above, the optimal path of the entire network topology is established. When a vehicle node receives/sends data packets, it implements the first step; otherwise, it will implement the second step.

4.2.2 Route development process

At the beginning, when the source node sends a data packet to the destination node, it checks the Q-table to see whether there is a next-hop node to the destination node. It then finds a neighbor node with the maximum Q-value to the destination node and forwards the data packet to it. If there is no such node, it starts the route discovery process. The source node

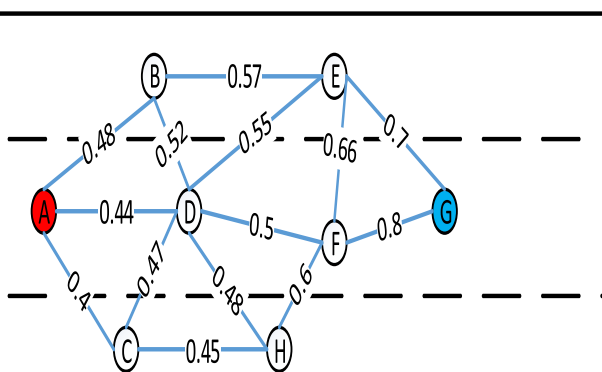


Fig. 4 Q-values to the destination node G as saved by each node

sends an R_req data packet by broadcasting to the entire network and starts a path request timer, where R_req records the IDs of all the nodes that it passed through in the routing process. When the destination node receives the first R_req packet from the source node, it saves the packet, and subsequently received packets will be discarded. From the R_req data packet, the destination node extracts the ID information that it recorded and flips it, then generates an R_rep data packet and writes the flipped path information into R_rep . After waiting for a time slot, it broadcasts the R_rep data packet to the nodes recorded in the flipped path. When a recorded node receives the data packet, it modifies the next-hop address, updates its Q-table, and sends out the packet through single-hop broadcast. Other non-destination nodes only modify their Q-tables and drop the packet when they receive it. Until R_rep is sent to the destination node, i.e., the source node of R_req , the source node cancels the request timer while updating its Q-table. At this time, a path has been found from the source node to the destination node, and the Q-tables of the nodes on the first path from source node to destination node are updated for the first time; their initial values in the Q-table are 0.

4.2.3 Route maintenance process

When the first route path is established, the Q-table values of some nodes adjacent to the path are also updated. To ensure that the path remains effective with dynamic changes in the network, it is important to start the route maintenance process. The main purpose of the route maintenance process is to maintain the Q-tables dynamically and to solve the network segmentation problem. Each node periodically broadcasts beacon packets to update the Q-tables of neighbor nodes, where the beacon data packet is mainly composed of the position, speed, and $\max(Q-Value)$ of the node. $\max(Q-Value)$ was described along with the learning process. To ensure update effectiveness, the transmission delay of the beacon packet in the experiment was set to a random number in the interval [0.5, 1]. The effective time of the destination node is specified for the Q-table. If the time of a destination node is longer than the specified time because it has not been updated, the destination node is considered invalid, and the corresponding column of data is deleted. When a network partition emerges due to vehicle movement, RSAR uses a store-and-forward strategy and starts the path request timer to broadcast an R_req data packet. If it does not receive the R_rep data packet sent from the destination node until the timer has expired, the destination node is considered to be unreachable, and the source node is informed to cancel the transmission; otherwise, the routing path will be reestablished at the breakpoint.

5 Experimental tests

NS-2 was used to verify the performance of the RSAR algorithm proposed in this paper. The simulation environment (scenario and parameter setting) will first be described, followed by a detailed analysis of the experimental results.

5.1 Simulation environment settings

NS-2 was used to simulate the experiment, and VanetMobiSim was used to generate a 1500×1500 square topology, as shown in Fig. 5. The topology consisted of intersections and straight roads, where each road was set to two two-way lanes, traffic lights were set at three intersections, and the change time of traffic lights was 5 s. To highlight the authenticity of the simulation, in the network environment, every vehicle node used an intelligent driving model (IDM), including changing lanes, overtaking, avoiding, and waiting. The generated mobility model was added to NS-2. Table 2 shows the basic NS-2 parameter settings. Over the whole network, 10 pairs of CBR data streams were randomly generated to send packets with a size of 512 bytes, and the transmission layer used UDP. The size of each beacon packet was calculated according to the transmitted information. To verify the performance of the RSAR algorithm effectively, two situations were set up to simulate the scenario in Fig. 5. In the first situation: the number of nodes in the entire network was set to a fixed value of 80, and the maximum speed of the vehicles ranged from 30 km/h to 90 km/h. In this situation, the discussion will focus on the effect of vehicle speed on the routing protocol. The second situation can be described as follows: the maximum speed of the vehicles was set to a constant 40 km/h, and the number of nodes varied from 60 to 120. In this situation, the discussion will focus on the effect of vehicle density on the routing protocol. At the beginning of the experiment, the nodes were randomly dispersed on different roads and moved on a fixed route. Each simulation time was set to 300 s, each situation was run 20 times, the average value was taken, and the simulation results were compiled.

5.2 Experimental results

The RSAR algorithm proposed in this paper was compared with the GPSR algorithm [17], the SLBF algorithm [11], the QLAODV algorithm [14], and the methods outlined in [22, 24, 26, 33, 37, 39]. These algorithms are all representative routing algorithms. This comparison makes it possible to evaluate the advantages and disadvantages of RSAR. In different scenarios, through comparisons of packet delivery ratio, end-to-end delay, and number of hops, the results shown in Figs. 5, 6, 7, 8, 9, 10, 11, and 12 were obtained. Figures 5, 6, and 7 show the comparisons in the first test case, and Figs. 8, 9, and 10 show the comparisons in the second one.

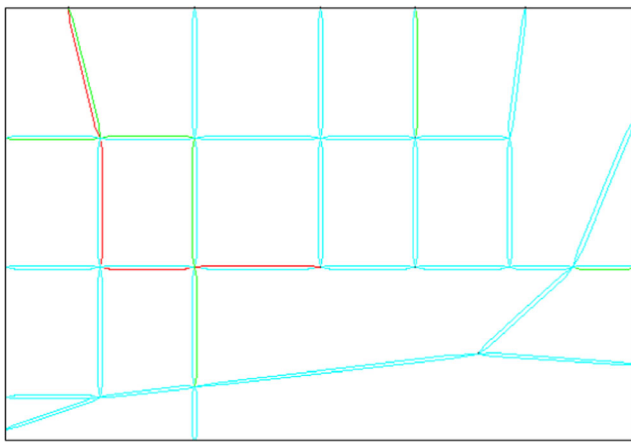


Fig. 5 Simulation topology

Figure 6 compares the relationship between packet delivery ratio and speed. The packet delivery ratio is the ratio of the number of data packets received by the destination node to the number of data packets sent by the source node. Figure 6 shows that, with increasing speed, the RSAR proposed in this paper achieved a high and stable packet delivery ratio, with an average ratio greater than 90%. The packet delivery ratios of the other three routing algorithms examined showed a sharp downward trend with increasing speed. This improvement was achieved because RSAR fully considered the influence of speed changes on link reliability. By evaluating the links between nodes, the reliability of these links was determined, and this information was used in routing decisions as a learning parameter of the Q-Learning algorithm. The packet delivery ratio of GPSR declined most quickly due to its selection of the next hop node without considering link reliability and due to its greedy mechanism. When the speed was less than 54 km/h, the packet delivery ratio of QLAODV was also greater than 90%, but when the speed exceeded 54 km/h, the ratio dropped sharply. Although it uses the Q-Learning model,

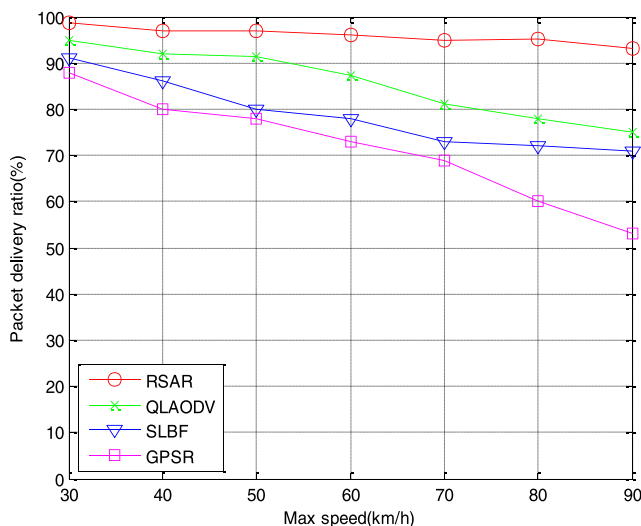


Fig. 6 Packet delivery ratio versus speed

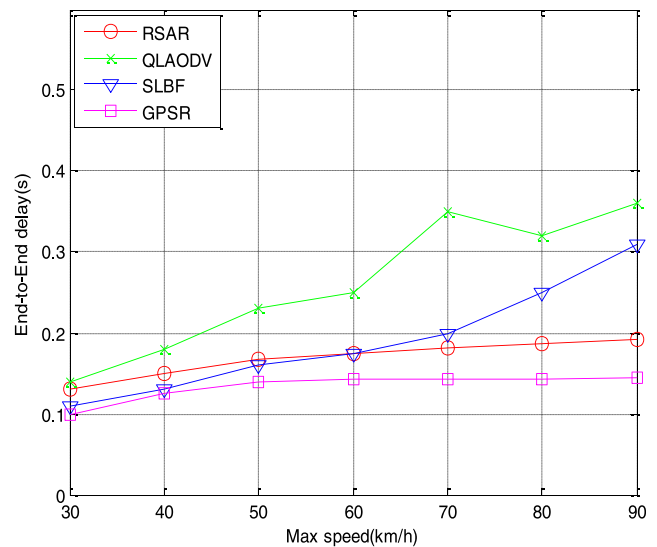


Fig. 7 End-to-end time delay versus speed

because of the need for maintenance of an end-to-end reliable path, QLAODV must repair the path continuously with increasing speed, leading to a decline in the ratio. The packet delivery ratio of SLBF also declined because it is affected by topology changes and link stability is not fully considered. Due to its use of a greedy approach, it was closer to the GPSR algorithm.

Figure 7 shows the relationship between end-to-end time delay and speed, where the time delay is calculated only as the average time taken by the destination node to receive valid data packets. Figure 7 shows that with increasing speed of the vehicle nodes, the time delays of the four algorithms all presented a rising trend. The time delay of the RSAR algorithm proposed in this paper was between those of GPSR and SLBF and was relatively stable. When the maximum speed exceeded 60 km/h, time delay of SLBF increased rapidly and became greater than that of RSAR. This trend was caused by the

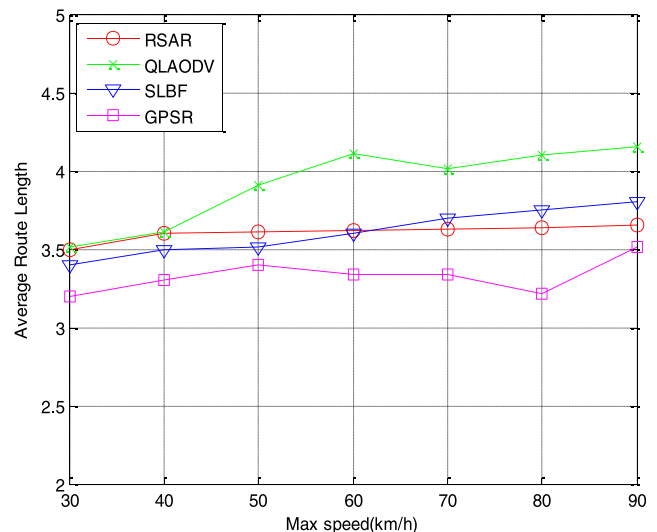


Fig. 8 Routing length versus speed

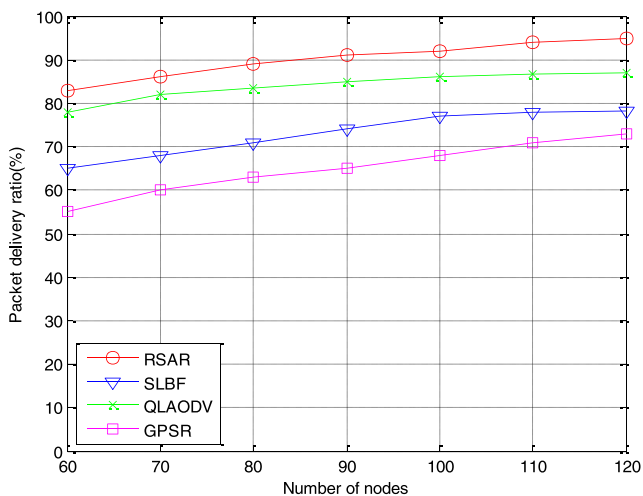


Fig. 9 Packet delivery ratio versus number of nodes

influence of topology changes and the need for recalculation of the effective forwarding area or re-transmission mechanism. Although RSAR was less affected by topology changes, the path with maximum Q-value is bound to have the largest bandwidth, the most reliable link, and the shortest routing length in this algorithm. The time delay of GPSR was the shortest because it only uses the greedy forwarding mode. The packet delivery ratio of GPSR had the lowest value because it only uses the greedy mechanism, so that it immediately drops packets when they fail to arrive. QLAODV also uses the Q-Learning model, but increasing speed made it switch paths frequently to maintain the effective routing path. This meant that the time delay of packet delivery was greatly increased. From Fig. 7, the packet delivery ratio of RSAR was between 0.1 and 0.2 under the rapidly changing topology environment. From Fig. 6, its packet delivery ratio was greater than 90%. It is necessary for these applications to transfer large amounts of data.

Figure 8 shows the relationship between average route length and speed, where the route length was calculated by

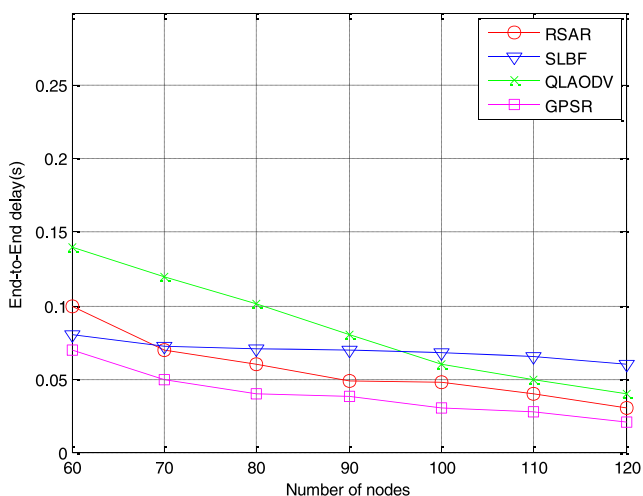


Fig. 10 End-to-end time delay versus number of nodes

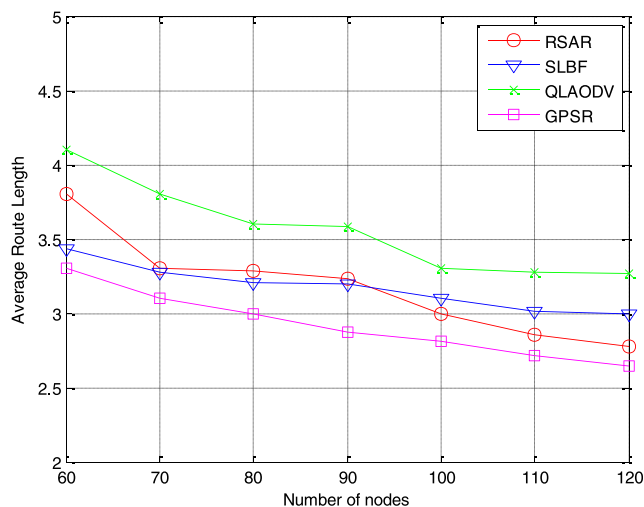


Fig. 11 Average route length versus number of nodes

the average number of hops taken for valid data packets to reach the destination. Figure 8 shows that the route length in RSAR was less than that of QLAODV because it does not use the path transformation mechanism to maintain the whole path, and the forwarding decision is used only to choose the node with maximum Q-value, so that its time delay is much shorter than that of QLAODV, as shown in Fig. 7. Figure 8 also shows that the RSAR proposed in this paper has a stable route length. This can be achieved because of the store-and-forward mechanism and the maximum Q-value selection mechanism, which ensures that each path selected is the shortest one. SLBF and GPSR both use a greedy approach to forward packets, but when the speed exceeds 60 km/h, under the influence of topology changes, the route length of SLBF increases; although GPSR does not use a reliable mechanism, the number of hops reached the minimum value.

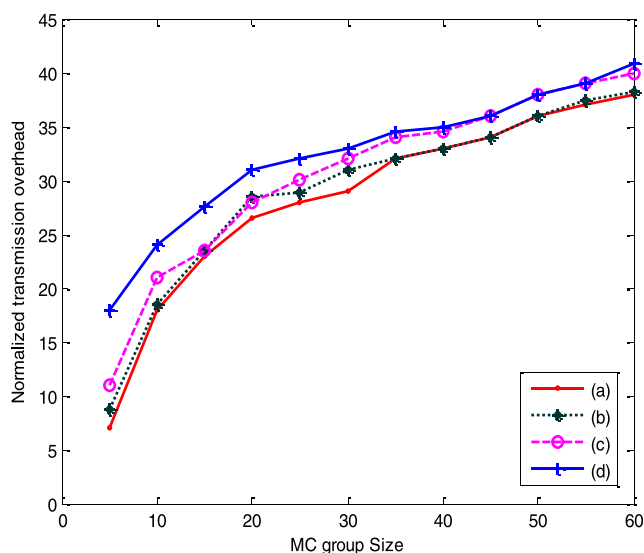


Fig. 12 NTO comparison among different methods: a the method proposed here, b [22], c [24], d [26]

Figure 9 shows the relationship between the packet delivery ratio and the number of nodes. As the number of nodes increases, the packet delivery ratio in all four algorithms shows a rising trend. When the number of nodes is 90, the packet delivery ratio of RSAR is greater than 90%, whereas that of QLAODV is 85%. Although these two algorithms are both based on the idea of Q-Learning, RSAR fully considers link stability between nodes so that the packet delivery ratio is greatly increased. Due to the store-and-forward mechanism, the packet delivery ratio of RSAR is much higher than that of SLBF and GPSR as the number of nodes increases. Because SLBF uses the retransmission mechanism, its packet delivery ratio is greater than that of GPSR; but because it is greatly affected by topology and the vehicle and the sending node are usually not on the same path, its packet delivery ratio is seriously affected and becomes lower than that of RSAR.

Figure 10 shows the relationship between end-to-end time delay and the number of nodes. The packet delivery ratio of RSAR gradually draws close to that of GPSR. The reason for this is that as the number of nodes increases, the number of learning nodes in RSAR also increases, so that the path to the destination node is shorter and the time delay is less. As the number of nodes increases, the time delay of QLAODV gradually draws close to that of RSAR, but with fewer nodes, it is much larger than that of RSAR. This behavior occurs because QLAODV spends much time on route maintenance. As the number of nodes increases, SLBF decreases slowly because it uses a timed broadcast mechanism that increases time delay. The time delay of all four algorithms shows a downward trend as the number of nodes increases.

Figure 11 shows the relationship between the average route length and the number of nodes. As the number of nodes increases, the average route length of all four algorithms shows a downward trend. This occurs mainly because the number of effective forwarding nodes increases. The length of the path found by RSAR is shorter than that of QLAODV. The path lengths of RSAR, SLBF, and GPSR are close because SLBF and GPSR both use a greedy approach. As the number of nodes increases, the next hop selected by RSAR is closer to the farthest node.

To show comprehensive comparative results for recent algorithms, extensive simulations and experiments have been conducted in this study to examine the performance of the proposed method. The Normalized Transmission Overhead (NTO) was defined to analyze the complexity of the proposed method and its cost. The metrics for the degree of complexity of the proposed method include context gathering overhead (O_{cg}) for VANET, transmission overhead (O_t) of data packets among VANET nodes, and others. According to classical computational complexity analysis methods [29–31], from the equations previously given in this paper, the degree of

computational complexity of the proposed method can be determined as $O(n)$:

$$NTO = \sum_{i=1}^{MC} \sum_{j=1}^{MC} \left(\xi O_{(i,j)}^{cg} + \varpi O_{(i,j)}^t + (1-\xi-\varpi) C_{(i,j)}^{de} \right), \quad (21)$$

where MC is the number of vehicle nodes in each evaluation group in VANET, such as $MC = 10, 20, 30, 40, 50$, or 60 , which means that there is a related number of vehicle nodes in a certain evaluation group in VANET. The integer parameters i and j are the node indices of each evaluation group in VANET. Parameters ξ, ϖ are real weighting values, $\xi, \varpi \in [0, 1]$ and $\xi + \varpi \leq 1$, such that the default real values are $\xi = 0.6, \varpi = 0.4$. $O_{(i,j)}^{cg}, O_{(i,j)}^t, C_{(i,j)}^{de}$ are the normalized real overhead values of vehicle nodes i and j for each evaluation group in VANET.

Based on Eq. (21), from the relative results comparison of existing methods and the method proposed here as shown in Fig. 12 ((a) the proposed method, (b) [22], (c) [24], (d) [26]), it is evident that the method proposed here has better performance than those proposed in Refs. [22, 24, 26]. This guarantees high QoS (such as lower overhead, less complexity, shorter delays, and transmission reliability) of VANET. Therefore, according to the above results, the method proposed here has certainly improved on routing overhead, transmission delays, rate of packet delivery, rate of losing packets, throughput, and other performance metrics for VANET.

In addition, many relative comparisons have been performed with the methods proposed in [33, 37, 39]. These relative comparison figures were not reported in this paper because the effects were similar to those shown in Fig. 12.

6 Conclusions

The routing service algorithm is a very important part of VANET. An efficient routing algorithm can greatly improve the data transmission rate, so that many applications can be run in VANET. To overcome the problems of dramatic topology changes and unreliable links caused by fast vehicle movements in VANET, a reliable adaptive routing service algorithm is proposed in this paper. First, the properties of vehicle motion and the factors that make links unreliable were studied, and the duration of a link was proved to have a log-normal distribution. Second, on the basis of this, a link reliability calculation model was developed. The link reliability between nodes was evaluated as a parameter and applied in the Q-Learning algorithm, following which the RSAR algorithm was proposed. Finally, simulation experiments were conducted to evaluate the performance of four routing algorithms, including packet delivery ratio, end-to-end time delay, and number of hops. The results showed that RSAR had a higher packet delivery ratio under various conditions and a low transmission time delay. RSAR can effectively solve the problems

caused by changes in topology through self-learning. However, because the learning process is a kind of local diffusion and there can be very many nodes participating in selecting the route, the routing expense in a large network environment will be enormous. Hence, future work by the authors will be focused on the problem of routing expense.

Acknowledgements This research work is supported by National Natural Science Foundation of China (Grant No. 61571328), Tianjin Key Natural Science Foundation (No.13JCZDJC34600), CSC Foundation (No. 201308120010), Major projects of science and technology in Tianjin (No.15ZXD5GX 00050), Training plan of Tianjin University Innovation Team (No.TD12-5016, No.TD13-5025), Major projects of science and technology for their services in Tianjin (No.16ZXF5GX00010, No.17YFZC GX00360), the Key Subject Foundation of Tianjin (15JCYB JC46500), Training plan of Tianjin 131 Innovation Talent Team (No.TD2015-23).

References

- Zhang DG, Li G (2014) An energy-balanced routing method based on forward-aware factor for wireless sensor network. *IEEE Trans Ind Inf* 10(1):766–773
- Zhang DG, Liu S (2017) Novel unequal clustering routing protocol considering energy balancing based on Network Partition & Distance for Mobile education. *J Netw Comput Appl* 88(15):1–9. <https://doi.org/10.1016/j.jnca.2017.03.025>
- Zhao CP (2012) A new medium access control protocol based on perceived data reliability and spatial correlation in wireless sensor network. *Comput Electr Eng* 38(3):694–702
- Seredynski, M., and P. Bouvry (2011) "A survey of vehicular-based cooperative traffic information systems". *Conference Record - IEEE Conference on Intelligent Transportation Systems*:163–168
- Al-Sultan S (2014) A comprehensive survey on vehicular ad hoc network. *J Netw Comput Appl* 37(1):380–392
- Zhang XD (2012) Design and implementation of embedded uninterruptible power supply system (EUPSS) for web-based mobile application. *Enterp Inf Syst* 6(4):473–489
- Johnson DB (2002) DSR: the dynamic source routing protocol for multi-hop wireless ad hoc networks. *Ad Hoc Netw*:139–172
- Abbasi IA (2014) A traffic flow-oriented routing protocol for VANETs. *EURASIP J Wirel Commun Netw* 2014(1):1–14
- Jerbi M (2009) Towards efficient geographic routing in urban vehicular networks. *IEEE Trans Veh Technol* 58(9):5048–5059
- Liu J (2015) A survey on position-based routing for vehicular ad hoc networks. *Telecommun Syst*:1–16
- Li C (2015) A self-adaptive and link-aware beaconless forwarding protocol for VANETs. *Int J Distrib Sens Netw* 2015(2):21–31
- Eiza MH, Ni Q (2013) An evolving graph-based reliable routing scheme for VANETs. *IEEE Trans Veh Technol* 62(4):1493–1504
- Yan G, Olariu S (2011) A probabilistic analysis of link duration in vehicular ad hoc networks. *IEEE Trans Intell Transp Syst* 12(4):1227–1236
- Toutouh J (2012) Intelligent OLSR routing protocol optimization for VANETs. *IEEE Trans Veh Technol* 61(4):1884–1894
- Zhu YN (2012) A new constructing approach for a weighted topology of wireless sensor networks based on local-world theory for the internet of things (IOT). *Comput Math Appl* 64(5):1044–1055
- Ke Z, Ting Z (2015) A novel multicast routing method with minimum transmission for WSN of cloud computing service. *Soft Comput* 19(7):1817–1827
- Song XD, Wang X (2015) New agent-based proactive migration method and system for big data environment (BDE). *Eng Comput* 32(8):2443–2466
- Zhang DG, Wang X, Song XD (2014) A novel approach to mapped correlation of ID for RFID anti-collision. *IEEE Trans Serv Comput* 7(4):741–748
- Wang X, Song XD, Zhang T (2015) New clustering routing method based on PECE for WSN. *EURASIP J Wirel Commun Netw* 2015(162):1–13. <https://doi.org/10.1186/s13638-015-0399-x>
- Zheng K, Zhao D-x (2016) Novel quick start (QS) method for optimization of TCP. *Wirel Netw* 22(1):211–222
- Wang X, Song XD, Li J, Chen YJ (2017) A kind of novel VPF-based energy-balanced routing strategy for wireless mesh network. *Int J Commun Syst* 30(6):1–15. <https://doi.org/10.1002/dac.2889>
- Zhu YN, Liu S (2016) Multi-radio multi-channel (MRMC) resource optimization method for wireless mesh network. *J Inf Sci Eng* 32(2):501–519
- Niu HL, Liu S (2017) Novel PEECR-based clustering routing approach. *Soft Comput* 21(24):7313–7323
- Hamed F (2018) Hybrid cost and time path planning for multiple autonomous guided vehicles. *Appl Intell* 48(2):482–498
- Zhou S, Tang YM (2018) A low duty cycle efficient MAC protocol based on self-adaption and predictive strategy. *Mobile Netw Appl* 23(4):828–839
- Ma Z (2017) Shadow detection of moving objects based on multi-source information in internet of things. *J Exp Theor Artif Intell* 29(3):649–661
- Chen J-q, Mao G-q (2018) Capacity of cooperative vehicular networks with infrastructure support: multi-user case. *IEEE Trans Veh Technol* 67(2):1546–1560
- Zhang DG (2012) A new approach and system for attentive mobile learning based on seamless migration. *Appl Intell* 36(1):75–89
- Zhang T, Zhang J (2018) A kind of effective data aggregating method based on compressive sensing for wireless sensor network. *EURASIP J Wirel Commun Netw* 2018(159):1–15. <https://doi.org/10.1186/s13638-018-1176-4>
- Zhang D-g, Ge H, Zhang T (2018) New multi-hop clustering algorithm for vehicular ad hoc networks. *IEEE Trans Intell Transp Syst* 7. <https://doi.org/10.1109/TITS.2018.2853165>
- Zhang T, Dong Y (2018) Novel optimized link state routing protocol based on quantum genetic strategy for Mobile learning. *J Netw Comput Appl* 2018(122):37–49. <https://doi.org/10.1016/j.jnca.2018.07.018>
- Asmae EG (2018) Energy efficient teaching-learning-based optimization for the discrete routing problem in wireless sensor networks. *Appl Intell* 48(9):2755–2769
- Sujoy R, Andrei S (2014) The multi-depot split-delivery vehicle routing problem: model and solution algorithm. *Knowl-Based Syst* 71(11):238–265
- Ma Z (2016) A novel compressive sensing method based on SVD sparse random measurement matrix in wireless sensor network. *Eng Comput* 33(8):2448–2462
- Liu S, Liu XH (2018) Novel dynamic source routing protocol (DSR) based on genetic algorithm-bacterial foraging optimization (GA-BFO). *Int J Commun Syst* 9. <https://doi.org/10.1002/dac.3824>
- Niu HL, Liu S (2017) Novel positioning service computing method for WSN. *Wirel Pers Commun* 92(4):1747–1769
- Ahmed LS (2017) An adaptive cooperative caching strategy (ACCS) for Mobile ad hoc networks. *Knowl-Based Syst* 120(15):133–172
- Chen C, Cui YY (2018) New method of energy efficient subcarrier allocation based on evolutionary game theory. *Mobile Netw Appl* 9. <https://doi.org/10.1007/s11036-018-1123-y>
- Liang JW, Ma MD (2018) A filter model for intrusion detection system in vehicle ad hoc networks: a hidden Markov methodology. *Knowl-Based Syst* 9. <https://doi.org/10.1016/j.knosys.2018.09.022>

40. Abboud K, Zhuang W (2014) Stochastic analysis of a single-hop communication link in vehicular ad hoc networks. *IEEE Trans Intell Transp Syst* 15(5):2297–2307
41. Liang YP (2013) A kind of novel method of service-aware computing for uncertain mobile applications. *Math Comput Model* 57(3–4):344–356
42. Song XD, Wang X (2015) Extended AODV routing method based on distributed minimum transmission (DMT) for WSN. *Int J Electron Commun* 69(1):371–381

Degan Zhang (M'01) Born in 1970, Ph.D. Graduated from Northeastern University, China. Now he is a professor of Tianjin Key Lab of Intelligent Computing and Novel software Technology, Key Lab of Computer Vision and System, Ministry of Education, Tianjin University of

Technology, Tianjin, 300384, China; He is a visiting professor of School of Electronic and Information Engineering, University of Sydney, Sydney, NSW 2006, Australia; His research interest includes service computing, etc.

Ting Zhang (M'03) Born in 1973, Ph.D. candidate, a Member (M) of IEEE in 2003. Now he is a researcher of Tianjin University of Technology, Tianjin, 300384, China. Her research interest includes WSN, etc.

Xiaohuan Liu (M'12) Born in 1988, Ph.D. candidate, a Member (M) of IEEE in 2012. Now she is a researcher of Tianjin University of Technology, Tianjin, 300384, China. Her research interest includes WSN, mobile computing, etc.