IJICTR
International Journal of Information & Communication Technology Research

Volume 12- Number 4 – 2020 (10 -25)

A Novel Protocol for Routing in Vehicular Ad hoc Network Based on Model-Based Reinforcement Learning and Fuzzy Logic

Omid Jafarzadeh
Department of Electrical, Computer and IT Engineering
Qazvin Branch, Islamic Azad University
Qazvin, Iran
omidjkh@gmail.com

Hadi Sargolzaey
Department of Electrical, Computer and IT Engineering
Qazvin Branch, Islamic Azad University
Qazvin, Iran
hadi.sargolzaey@qiau.ac.ir

Mehdi Dehghan*
Department of Electrical, Computer and IT Engineering
Qazvin Branch, Islamic Azad University
Qazvin, Iran
dehghan@aut.ac.ir

Mohammad Mehdi Esnaashari
Faculty of Computer Engineering
K. N. Toosi University of Technology
Tehran, Iran
esnaashari@kntu.ac.ir

Received: 26 September 2020 - Accepted: 2 December 2020

***Abstract*—Vehicular ad-hoc networks (VANETs), as a result of today's vehicles equipped with different wireless technology, have been attracting interest for their potential roles in many fields such as emergency, safety, and intelligent transport system. However, the development of a reliable routing protocol to route data packets between vehicles is still a challenging task due to the high mobility, lack of fixed infrastructure, and obstacles. One technique to tackle this challenge is using machine learning. In this paper, we have proposed a protocol applying multi-agent reinforcement learning (MARL) as a technique that enables groups of reinforcement learning agents to solve system optimization problems online in dynamic, decentralized networks. Our protocol is based on a model-based reinforcement learning method which has a higher convergence speed compared to the model-free one. To form the needed model for MARL, we have developed a Fuzzy Logic (FL) system that evaluates the quality of links between neighbor nodes based on parameters such as velocity and connection quality. The performance of the proposed protocol is studied by extensive simulation with respect to various metrics such as delivery ratio, delay, and overhead. The results obtained show significant improvement of VANETs performance in terms of these metrics.**

***Keywords*—VANET; Routing; Reinforcement Learning; Fuzzy Logic**

* Corresponding Author

I. INTRODUCTION

A Vehicular Ad hoc Network (VANET) is a kind of wireless network with different applications such as safety, entertainment, emergency, and so on. However, the main reason for researchers work on such networks is their essential role in constructing a proper framework for intelligent transportation systems. There are usually two kinds of connections in a VANET as shown in figure 1:1) multi-hop Vehicle to Vehicle (V2V) connection with no infrastructure and 2) vehicle to Road Side Unit(RSU) connection through which a vehicle can access other infrastructures (like the Internet). VANET nodes are vehicles with dynamic characteristics (such as speed, acceleration, direction, etc.) that move in today's natural urban environment with special features (buildings, overpasses, fixed roads, etc.) blocking signal transmission. Therefore, creating and maintaining a stable path between a source and destination pair without any degradation or loss of quality has always been a challenging task. This problem is more evident in V2V connections. Creating stable routes through different routing techniques is studied and investigated widely in previous research projects [1-6]. Most of these techniques are classified into either topology-based or position-based[7]. In topology-based techniques, an information link is used to deliver data packets from a source node to a destination node. In the position-based techniques, each node is aware of its geographical location as well as others' (via GPS, digital map, etc.) and utilizes such information for routing. Since position-based algorithms do not utilize any routing table, they are better suited for the highly dynamic nature of VANET[8]. One of the essential considerations that must be taken into account for presenting a proper routing algorithm is that it should be adaptable with continuous and unpredictable changes of network topology in VANET. Many researchers have tried to meet this requirement via applying different mechanisms, such as maintaining routes in proactive protocols, using periodic updates in reactive approaches, or using link stability metrics when constructing paths[9]. However, many of these protocols have simple assumptions for network attributes or consider models for wireless channels, which are not always true, especially for VANET. It is also seen in some approaches that several parameters are simply set with a predefined threshold while they may be dependent on network situations.

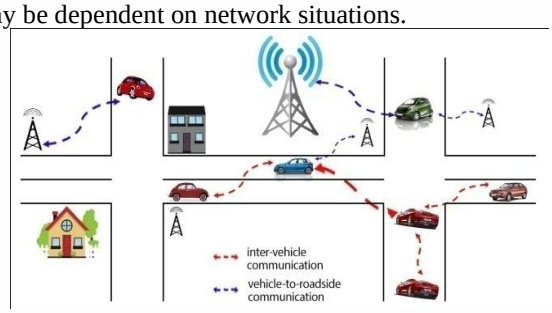


Figure 1. VANET communication types.

However, our proposed Reinforcement Routing Protocol for VANET (RRPV), attempts for desirable adaptability for routing in VANET using a combination of model-based reinforcement learning and Fuzzy Logic(FL). RL is an unsupervised learning technique that enables an agent to monitor the state of its environment and doing an action that effects its environment in order to learn an optimal policy. By an optimal policy for routing in VANET, we mean selecting the best neighbor as the relay node during packet forwarding. RRPV is based on the multi-agent reinforcement learning (MARL) technique. MARL corresponds to a learning problem in a multi-agent system in which multiple agents learn simultaneously. In RRPV, all nodes of the network are considered agents that cooperate together to find the best path(optimal policy). Using a model, the agent can predict the resultant next state and next reward after doing an action in a state. Having a model, the agent will apply a computational process that takes a model as input and produces or improves a policy. Our main contribution in contrasting the model for RL is using a Fuzzy logic system. The FL is utilized to confront uncertainty in link quality.

To sum up, our main contributions in this paper are as follows:

1) Developing an FL system for evaluating the qualities of links between vehicles and deriving the desirable model for RL.

2) Proposing an algorithm applying Reinforcement Learning (RL) in choosing proper neighbors for relaying packets towards their destinations.

The remainder of this paper is organized as follows: Section 2 is dedicated to a review of related work. The System model is described in Section3. Basics of RL, Multi-Agent Reinforcement Learning (MARL), and FL system are described in Section4. The proposed protocol for routing is presented in Section5.In section 6, simulation and evaluation of the proposed approach are presented. Finally, Section 7 concludes the paper.

II. RELATED WORK

Many protocols have been designed for routing in VANET so far. These protocols are usually differentiated based on network infrastructure and used techniques. Generally, VANET routing approaches are categorized into two groups. The first group is topology-based protocols that use the information of links for forwarding packets. These approaches are implemented in two ways: proactive and reactive [6]. In proactive ones, routes are formed based on shortest-paths algorithms and stored in tables and then are used whenever they are needed. However, in reactive ones, the routes are formed only when it is requested, and just active paths are maintained in tables. Most of the topology-based routing protocols in VANET are originated from Mobile Ad hoc Network(MANET)[10]. In this regard, some researchers have tried to apply MANET routing protocols for VANETs.

The approaches proposed in[11] are some examples of this effort. Ad hoc On-demand Distance Vector(AODV) routing protocol[12] as a famous

MANET routing protocol is changed to be used in VANET. In this work, two algorithms are proposed: (1) connection-based restricted forwarding (CBRF) and (2) two-phase routing protocol (TOPO). CBRF is usually used in small networks, while TOPO is more suitable for large networks. However, according to the experiments done in[13-16], the use of topology-based protocols for routing in VANET, with more dynamic properties than MANET, is not effective in terms of performance metrics. Especially, these protocols usually impose a further overhead burden on networks because of their activities to maintaining and discovering paths between sources and destinations. As a result, they will face scalability challenges.

The second group of routing approaches in VANET includes Geographic-based protocols. These protocols use the information of network positions that are obtained from a digital map of streets, traffic models, or location systems. By considering the movement limitation of vehicles on the road surfaces and also excessive use of GPS, it seems that these types of approaches can be more efficacious [4,3]. Greedy Perimeter Stateless Routing (GPSR)[17] was one of the first proposed protocols in the Geographic-based group in which, first, a source node obtains the location of a destination and then, by using a greedy algorithm, tries to select the neighbors that have the least distance to it.

In [18], a position-based scheme is proposed with the main goal of better video transmission. Here, besides main routes, some independent paths are also founded between a pair of source and destination and then used whenever they are needed. It also develops a closed equation for estimating link probabilities.

Huang and Lin[19] proposed a promoter algorithm in which each node selects the furthest vehicle to forward the packets. Abuashour and Kadoch[20] have also proposed a Geographic-based protocol that uses the velocity of vehicles as the main metric for determining link stability. The basis of their work is clustering networks and sending data via the heads of each cluster. However, forming and maintaining clusters in VANET can result in higher overhead due to the frequent changes of node position and high dynamics. Generally, most of the protocols we have mentioned fall in conventional and computational categories of protocols that try to find an optimal solution for routing using mathematical methods and pure theory. However, these approaches are not sufficient for large-scale VANET with highly dynamic properties[21]. Nowadays, another group of techniques is proposed for routing, which use Artificial Intelligence (AI) schemes[22-26]. They aim to enhance the ability of algorithms under continuing and unpredictable changes of VANET network topology via learning techniques.

Situation-Aware Multi-constrained QoS Routing Algorithm (SAMQ) [27] is one of these approaches in which an effort was made to present a situation-aware multi-constrained Quality of Service (QoS) routing protocol by applying the concept of situational awareness and an Ant-colony based Algorithm (ACO). In [28] also a technique with a learning scheme is proposed in which the appropriate intersections for transferring data are selected, and routes are created by considering QoS limitations. These limitations are based on three metrics, including packet delivery ratio, delay, and connection probabilities. The problem of choosing a path is mathematically formulated as an optimization problem, and then, an ACO-based algorithm is proposed to solve it. Then a local QoS model is also offered for each part of a city to reduce the traffic overhead.

A routing protocol based on RL is also proposed in[29] by considering the effect of the transmission rate of the MAC layer in selecting links to construct routes. The proposed scheme is comprised of two sections. At first, it introduces an algorithm based on Q-learning to estimate the transmission rate at the MAC layer, and for this, it tries to find a relation between hello reception ratio and best MCS (modulation and coding scheme). In the second part, another algorithm is proposed, which is again based on Q-learning for routing and selecting the best neighbor to forward packets toward destinations. Node selection is made according to the action-value function, which is stored in Tables. By considering network lifetime maximization and delay minimization as the quality of service constraints of the routing problem, a micro-artificial BEE colony-based solution has been proposed to address this issue using a multicast routing scheme in[30] . The work of [31], is one of the newest efforts to address the routing problem in VANET. It first calculates the reliability of the inter-node link by analyzing the characteristics of the vehicle movement. Then this parameter is used in the improved Q-learning strategy. It introduced two heuristic functions. The first function is used to speed up learning, and the second one is used to reduce unnecessary exploration. The work of [32] is the other efforts for routing problem in VANET, which tries to predict the destination and movement patterns of each node by using forward and backward technique of Hidden Markov Model (HMM), and then during the neighbor selection process, it selects nodes that have the most chance to deliver a packet to the final destination. Another protocol that apply reinforcement learning for routing is proposed in[33]. It is a combination of Q-learning and grid-based routing. It works in two parts: first, it divides the area into grids and finds the next optimal grid toward the destination based on the Q-value table. In the second part it uses a greedy selection algorithm to choose the nearest neighbor towards the destination.

III. System Model

Our system is a network of *n* mobile nodes formed from vehicles. Vehicles are assumed to move with a constant velocity in two directions of roads. Each vehicle is equipped with an On-Board Unit (OBUs). It also has a Global Positioning System (GPS) receiver, which indicates its location as well as velocity. A digital map is also available in each node for obtaining the geographical location of destination nodes via location services. Vehicles communicate with each other only in an ad hoc mode by applying the IEEE 802.11p protocol. All nodes have the same transmission range. Civil

buildings and structures are considered as obstacles that affect the communication in VANET due to the use of a high-frequency band above 5.8GHz. For example, it is likely nodes that are physically adjacent, cannot communicate with each other due to the presence of a building between them.

IV. REINFORCEMENT LEARNING

Reinforcement learning means selecting the best action by an agent according to its current situation in the environment. The best action is found using a series of trial and error throughout the environment. The environment is modeled as a set of states and transition probabilities between these states. The agent can select one action among a set of actions in each state. A policy, π, is utilized by the agent for selecting an action among the list of available actions. An agent will receive a reinforcement signal from its environment by performing an action in each state. The received signal will be used to update the policy of the agent using a learning strategy. Here, the aim is to determine how an agent can change its policy by benefiting from its experiences so that the obtained reward would be the most in the long run.

The reinforcement learning problem is usually modeled as a Markov Decision Process(MDP)[34]. An MDP is formed of a set of states $S=\{s_1, s_2, \ldots, s_n\}$, a set of actions $A=\{a_1, a_2, \ldots a_n\}$, a reward function $R:S\times A \longrightarrow R$ and a state transition function $P: S\times A \rightarrow \Pi(S)$where $\Pi(S)$ is a set of probability distributions.

A. Value Function

The value function is a key component in any reinforcement learning algorithm. It is a function of states or a pair of state-action. This function determines an estimation of income for an agent to being in a state or doing an action in a state following a policy like π. Value functions are usually defined in two forms as follows:

$$V_\pi(s)=E_\pi[G_t|S_t=s] \quad (1)$$

$$Q_\pi(s,a)=E_\pi[G_t|S_t=s,A_t=a] \quad (2)$$

In equation (1) and equation(2) $E[.]$ is the expected value that an agent can earn by policy π, and t is a timestamp. G_t is a function of the reward sequence. $V(s)$ and $Q(s,a)$ are called *state-value* function and *action-value* function, respectively. In this paper, we refer to them as *v-value* and *q-value* in short.

B. Model-based reinforcement learning

While in model-free RL, the optimal policy is approached through the interaction between an agent and its real environment, in the model-based RL, first, an internal model of the environment is constructed, then an optimal policy is calculated based on that model. We can refer to Q-learning[35] as an example of model-free methods in RL. However, these methods are usually slow in finding the optimal policy. As a result, these methods are not suitable for the highly dynamic environment of VANET. On the other hand, while model-based approaches are better suited for these networks, it is required to form a dynamic state transition model and sometimes a reward model before applying such approaches. We will back to this in Section 5, where we describe entirely how to form such models for the routing problem in VANET.

C. Multi agent reinforcement learnin(MARL)

In MARL, besides local learning, information and observation that are obtained locally in each agent are exchanged between neighbors. In fact, cooperation is formed between nodes to acquire a global optimum. This scheme also helps anode to consider the performance of its neighbors, and hence prevent it from becoming a selfish node, especially in a wireless network with hared transmission media. As is shown in figure 2, MARL divides a network-wide problem into some components, each of which is solved by a self-organized agent. We consider the routing in VANET as a Discrete Optimization

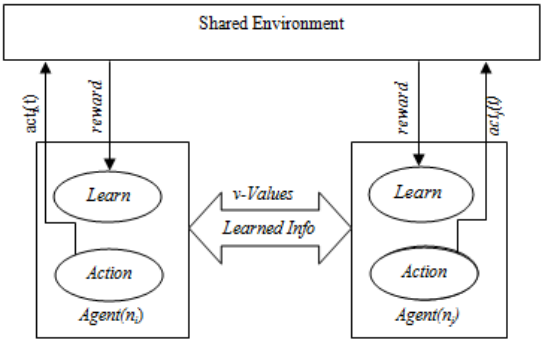


Figure 2. MARL agents.

Problem(DOP)[36] that should be solved by collaborating RL agent. To address this problem, we use the MARL technique in which each node of a network is considered as an agent that shares its information about the environment with others in the form of *v*-Values exchanges. The solution to each DOP is initiated at some starting agent in the network and terminated at some (potentially remote) agent in the network.

D. Fuzzy logic

One of the main attributes of the VANET that raises some challenges for routing is its inconsistency. As we know, a human can decide in different situations even when there is scant and uncertain knowledge. Here, we aim to give this ability to a routing system for VANET by applying FL. This logic was introduced for the first time by Lotfi-Zadeh[37,38] as a tool for working with uncertain data. The fundamental concept of this logic is making a fuzzy set that is against classic set theory with zero-one logic for membership. Membership in an FL set is ranked from zero to one. The inference is made by using fuzzy rules, after forming sets. These rules are usually presented in the form of *if-then* statements. There are several ways for inference in FL; the most well-known one (which is also used in this paper) is the Mamedani method[39]. In this type of deduction, it is said that if x is A and y is B then z is C. Output of the inference phase in FL is also presented as fuzzy sets but in a real work system, real numbers are used. As a result, for the final utilization of fuzzy inference, the output should be converted to real numbers via a defuzzification process.

V. PROPOSED MODEL

Our main contribution in this paper is using Dyna-Q[40] architecture to integrate the major function needed in an on-line planning agent. The architecture has two parts: model learning and reinforcement learning which should be occurred simultaneously. We will use a fuzzy model (developed in the next section) for the first part, and an MDP planner for the second part. The overall architecture of this scheme is depicted in figure 3. The central column represents the fundamental interaction between agent and environment that leads to real experiences. The arrow on the left represents the direct reinforcement

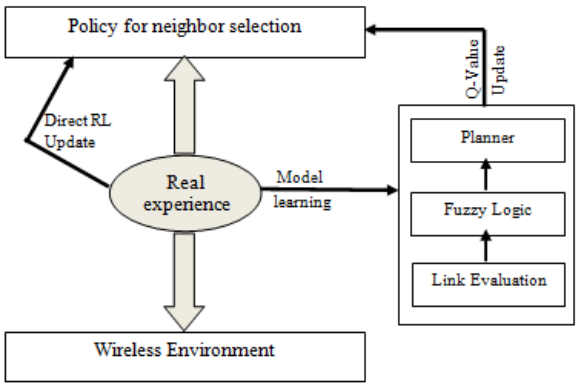


Figure 3. The proposed scheme.

learning that is operated on real experience to improve policy(value functions) for neighbor selection. On the right are model-based processes in which the following operations are done:

1. The gathered information from the environment is used to compute quality parameters of links between neighbors.
2. The fuzzy system then evaluates the links based on the computed parameters. This evaluation is done using fuzzy rules. Results of link evaluation are fed as state transition probabilities to the next step, which is an MDP planner.
3. MDP produces *q-values* that are used for selecting neighbors as relay nodes.
4. According to the policy formed by MDP, relay nodes are selected for each received packet.

A. Fuzzy logic for link evaluation

It is our primary purpose to apply a model-based RL for routing problem. By a model of the environment, we mean anything that an agent can use to predict how the environment will respond to its action. However, the main requisite during applying model-based schemes, is determining a model for a state transition and also a reward. A node of VANET that has a packet for forwarding toward the destination is considered as an agent that will change its state via delivering a packet to one of its neighbors. So, for determining state transition, we should specify the probability with which a packet is transmitted by a node and reaches an adjacent node. This probability is highly dependent on the quality of the link between the two adjacent nodes. The fact that a link is qualified enough or not depends on many parameters such as bandwidth, movement directions of the two nodes, relative velocity, received signal strength, and so on. Since these factors are dependent on the environment, so having a mathematical model for deriving an optimal solution increases the complexity of the algorithm, and the designed model will not have the flexibility that is required in VANET. Thus, we plan to solve this problem using a fuzzy system. The main attribute of an FL system is that it helps decision-making in an unsure environment with uncertain and estimated information.

B. Parameters used for link evaluation

As mentioned earlier, transition probability describes the probability with which a packet is transmitted by a node, n_i, and reaches an adjacent node, n_j. This probability is related to the quality of links that connect these two nodes. Thus, we will use the following parameters in FL system to evaluate the quality of these links.

1) Link stability

The stability of a link between two nodes n_i and n_j indicates how long the connection between them is available. We show this factor as $stab(i,j)$. The primary component in calculating this factor is the relative velocity and movement directions of the two nodes:

$$stab(i,j)=\begin{cases}\dfrac{T_e}{M} & if\ T_e \le M \\ 1 & otherwise\end{cases} \quad (3)$$

Where, T_e indicates the estimated time that the connection between the two nodes remains and M is a constant value that is determined based on the simulation time. T_e is calculated in two situations as follows:

i) Two nodes *i* and *j* move in the same direction

$$T_e=\begin{cases}\dfrac{R+d_{ij}}{|v_i-v_j|} & if\ v_i > v_j \\ \dfrac{R-d_{ij}}{|v_i-v_j|} & if\ v_i < v_j\end{cases} \quad (4)$$

ii) Two nodes *i*, and *j*, move in the opposite direction

$$T_e=\begin{cases}\dfrac{R+d_{ij}}{|v_i+v_j|} & getting\ \ close \\ \dfrac{R-d_{ij}}{|v_i+v_j|} & getting\ \ away\end{cases} \quad (5)$$

Where R is the transmission range of a node and d_{ij} is the Euclidean distance between two neighbors.

We rank the stability of a link in three sets: low, medium, and high. Based on the above equations, the membership function for the stability factor is defined in figure 4.

2) Connection quality

We also consider the connection quality while evaluating the quality of a link. Estimating a precise metric for the quality of a connection in a dynamic network such as VANET is usually a difficult task. We have used the ratio of sent/received of hello packets for this purpose, as indicated in Equation(6).

$$QoC(i,j) = \begin{cases} \frac{Rec(i,j)}{Sen(i,j)} & if \ \ T(i,j) \geq P \\ \frac{Rec(i,j)}{Sen(i,j)} \times \left(1 - (\frac{1}{2})^{S(i,j)}\right) & if \quad T(i,j) < P \end{cases} \quad (6)$$

An interval P is considered for evaluating the QoC factor of the link (i,j). The value of this factor is updated in each interval based on Equation(6). Here $S(i,j)$ is the number of hello packets sent from node n_i to a neighbor node n_j in a specified interval, and $Res(i,j)$ indicates the number of received hello packets by node n_j. $T(i,j)$ is the duration within which the two nodes have been neighbors. Hello packets are usually sent within a specified time interval during P (e.g. P=1s).So when P is 10s, $QoC(i,j)$ will be calculated as 0.8 with $T(i,j)$>10s, $Sen(i,j)$=10 and $R(i,j)$ =8. Now suppose we have another link with $T(i,k)$=2, $Sen(i,k)$=2, and $Rec(i,k)$=2.Then $QoS(i,k)$ is calculated as 1, which indicates that the link (i,k) has better quality than link (i,j). However, this is a false evaluation because a longer $T(i,j)$ increases the chance of collision and packet loss. Thus, for a true evaluation, we have considered a discount factor for nodes whose $T(i, j)$ is less than P in Equation(6).

Based on the above equations, the membership function for the connection quality factor is defined in figure 5.

C. Mapping and inference rules

Calculated factors in the previous section are used by each node to evaluate the links between themselves and their neighbors using *if-then* rules presented in Table1.

TABLE I. INFERENCE RULES

	Stability	Quality	Status
Rule1	high	good	excellent
Rule2	high	medium	good
Rule3	high	bad	poor
Rule4	medium	good	good
Rule5	medium	medium	acceptable
Rule6	medium	bad	bad
Rule7	low	good	poor
Rule8	low	medium	bad
Rule9	low	bad	very bad

Linguistic variables used to describe a link have values like *excellent*, *good*, *acceptable*, *poor*, *bad*, and *very bad*. Since several rules may be activated simultaneously, the Max-Min method[41] is used to combine evaluation results.

D. Output

After evaluating a link between two nodes based on fuzzy sets, a real number should be generated to indicate the quality of the link. This number is generated based on the defuzzification process and output membership function. There are different approaches to defuzzification. Here, we use a center-of-gravity method based on an output membership function that is depicted in figure 6. The output of this part is a real number between 0 and 1. We show this number by $Z(i,j)$, which is an evaluation of the link between two nodes n_i and n_j. These values are used as the state transition values, which are required in Equation(7), which will be discussed in the next section.

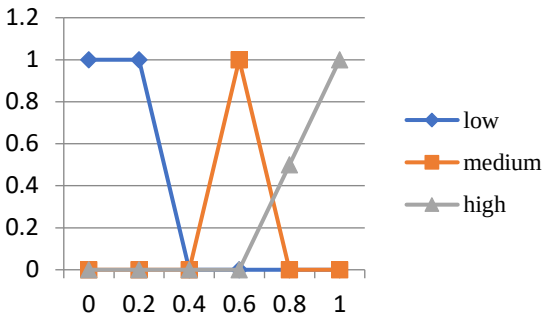


Figure 4. Membership function for stability factor.

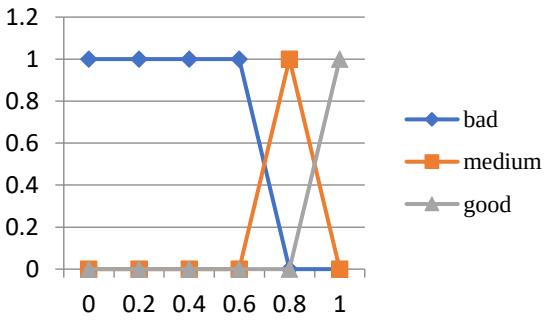


Figure 5. Membership function for connection quality factor.

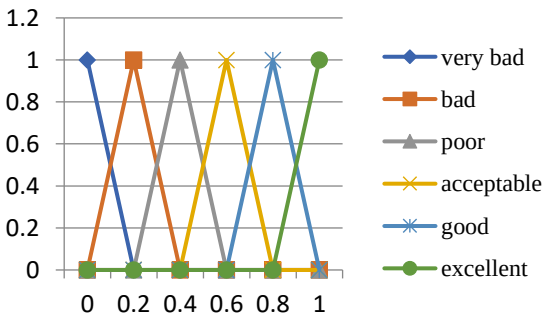


Figure 6. Output Membership function.

E. Routing algorithm

Each node uses a routing table in which the *q-values* of itself and *v-values* of neighbors are stored. Values of this table are computed based on the evaluation of all links between node n_i and its

neighbors by using the model described in Section5.2. Next hop selection for forwarding a packet is done based on the policy π, which is formed via the soft-max scheme[40]. The *v-value*s are also updated periodically by receiving a new advertisement over time. At the beginning of the activity time of the network, when no routing information is available, and tables are empty, nodes use broadcast messages to discover routes. The details of our algorithm are as follows:

1) Each vehicle is considered a node of the network with two independent states; *F* and *D*, where state, *F* indicates that the node has a packet for forwarding and state *D* shows that the packet is delivered to one of the neighbors. Our RL task here is episodic. Each episode in each node is started when a packet is available to be forwarded (or to be sent). Every episode is started at state *F*. The episode will be finished when the packet is sent. Both *F* and *D* are final states. *F* is the final state of an episode if, in that episode, the packet was not delivered successfully and *D* is the final state of the episode when the packet was successfully delivered to the selected neighbor (see figure 7).

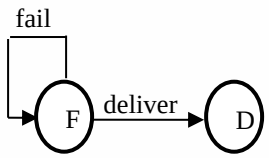


Figure 7. RRPV MDP with two states.

2) A set of actions is assumed for each node n_i as $A_i = \{a_1, a_2, \ldots a_m\}$ in which a_j indicates a transmission from the current node n_i to the neighbor node n_j and *m* is the number of neighbors of a node. Each node uses periodic hello messages to form this set.

3) There are two state transitions for each node n_i: 1) *F=>D* that determines a successful packet delivery to a neighbor n_j and 2) *F=>F* that indicates failure of this delivery. The probabilities of these transitions are shown by $P(D|F,a_j)$ and $P(F|F,a_j)$, respectively,

where a_j shows the action of choosing n_j by n_i as the next hop of the packet. The values of these state transition probabilities are determined according to the result of fuzzy inference (the value of $Z(i,j)$ that was calculated in the previous Section) as follows:

$$P(D|F,a_j)=Z(i,j) \text{ and } P(F|F,a_j)=1-Z(i,j) \qquad (7)$$

4) Each node n_i uses estimated *v-values* of its neighbors when calculating the action-value function $Q(s,a)$. Neighbors are informed about the changes of *v-values* via piggybacked advertisements in data packets. This way, a node can acquire a local cache about its neighbors. The cache contains a table of *q-values* for each action (selecting a neighbor node n_j towards destination *dest*) and also the last advertised *v-value* from node n_j for successful transmission to that destination. The names and contents of each field of this table are shown in Table 2. The entry $V_{j,dest}$ in the table, indicates *v-value* received from a neighbor node n_j for destination *dest.* $Q_{dest}(s,a_j)$ shows the computed action-value function in n_i by selecting node n_j as the next step of packets towards the destination *dest*. By receiving each packet in node n_i, it evaluates all links between itself and neighbors by using the model described in section 5.2. Then it calculates (or updates) the $Q_{dest}(s,a_j)$ values of the table, and finally, it selects one of the neighbors based on its policy. The value of $V_{j,dest}$ is also updated periodically by receiving a new advertisement over time.

TABLE II. ROUTING TABLE CONTENTS

Field name	Field Content
row	The id of neighbor node n_j
dest	destination node n_{dest} is reached from current node n_i via neighbor node n_{row}
$V_{j,dest}$	Estimated route cost from node n_{row} to destination node n_{dest}
$Q_{dest}(s,a_j)$	Action-value function for current node n_i for selecting neighbor n_{row} as the next hop toward the destination node n_{dest}

5) Each node n_i has a specified policy π_i, during which the probability of selecting each of the neighbors as the next hop in reaching the destinations is determined. To form this policy, we have used Boltzmann distribution, as a common softmax method, over *q-values*:

$$P(F,a_j)=\frac{e^{-\left(Qdest(F,a_j)\right)/\tau}}{\sum_{a_j} e^{-\left(Qdest(F,a_j)\right)/\tau}},\ a_j \in A_i \qquad (8)$$

We have chosen softmax for selection because, in a highly dynamic environment, the selected strategy should be able to make a proper balance between exploitation and exploration. The soft-max rule is one way to control the relative levels of exploration and exploitation. The factor τ, which is named temperature value, sets the required balance between them. A value of low temperature $(\tau \to 0)$ will propel the process of action selection towards a greedy exploitive scheme. As the value of the τ is increased, the chance of finding a more optimal path is also increased (exploration scheme). Generally, when the network is highly dynamic, and the opportunity of finding a stable path is low, the temperature value should be set with higher values.

6) In an MDP, each agent will receive a reward from its environment by doing an action. We consider two parameters for the reward model. First, the quality of the link between node n_i and the selected next-hop (e.g. n_j), and second, the distance between node n_i and n_j. As a result, the immediate reward of doing action a_j by node n_i can be stated as equation (9):

$$R_{i,j}= \alpha Z(i,j) + \beta W(i,j) \qquad (9)$$

Where, α and β are normalization factors and $W(i,j)$ is a distance factor that is normalized and determined as follows:

$$W(i,j)=\left(\frac{1}{L}\right)*T(i,j) \qquad (10)$$

Where, $T(i,j)$ indicates the Euclidean distance between the node n_i and n_i and L is the largest possible distance between two neighbors.

7) Given state F at node n_i, the *q-values* will be updated based on a distributed model according to a reinforcement learning algorithm[23]:

$$Q_{dest}(F,a_j) = P(D \mid F,a_j)*\left(R(s' \mid F,a_j)+V_{j,dest}(D)\right)+ P(F \mid F,a_j)*\left(R(s' \mid F,a_j)\right), s' \in \{F,D\}, \quad a_j \in A_i \qquad (11)$$

Where, $V_{j,dest}$ is the *v-value* of node n_j and $R(s'|F,a_j)$ is the immediate reward obtained by doing action a_j in state F and is computed as:

$$R(s'|F,a_j) = \begin{cases} R_{i,j} & if \;\; s' = D \\ -1 & else \end{cases} \qquad (12)$$

Where $R_{i,j}$ is derived from Equation(9)

Then, each node will calculate the *v-value*, V_i, using Bellman Equation[42] as follow:

$$V_{i,dest}(D) = \max_{a_j \in A_i} \left[Q_{dest}(F,a_j)\right] \qquad (13)$$

8) The *v-values* calculated by equation (12), are advertised in the network through piggybacking in data packets. It should be mentioned that in equation (10), we do not discount the acquired reward in the future.

9) The routing table is updated periodically, and the routes that are not used for a specific, are gradually degraded, and finally removed from the table. Each node periodically updates the value of *v-value* received from neighbor node n_j, (V_j),for a destination as follows:

$$V_{j,dest} = V_{j,dest} * \gamma^{te} \qquad (14)$$

Where, the value of *te* is the elapsed time of the last received advertisement from neighbor node n_j and γ is a number between 0 and 1 which determines the degradation rate.

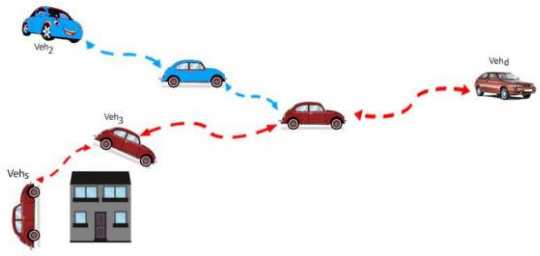
The used algorithm by each node n_i for routing is presented in Algorithm 1:

1: production of transition model:
Evaluate the links between neighbors;
Apply fuzzy logic system developed at section 5;
Evaluate the values of z(i,j) as transition model;
2: Forming the policy π_i for selecting next hop toward destination:
double R[ ],temp=0;
for(each row j in routing table)
If(dest==P.dest)
{
Compute P(F,a_j) using equation(8);
R_j=temp+ P(F,a_j);
temp=temp+ P(F,a_j);
}
3: Randomly select an action a_j according to the probability distribution:
Generate a random number 0≤ $\mathcal{E}$ ≤1;
for(each R_j)
If(R_{j-1}<$\mathcal{E}$ <R_j)
{
deliver packet P to n_j;
}
4: Calculate $R(s'|F,a_j)$ as the immediate reward using equation(12)
5: Updating the q-values and v-values
For each state S of node n_i and $a_j \in A_i$, update q-value Q(S,a_j) using equation(11);
Compute or update v-value for the destination node n_{dest} using equation(13);
Advertise v-value;

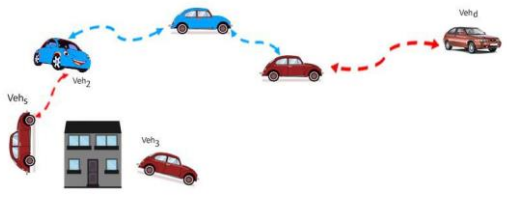
Algorithm 1: the model-based reinforcement learning algorithm at node n_i

F. Sample scenario

As an example of the ability of the proposed protocol in adapting to a dynamic environment, consider the scenario depicted in figure 8(a) and figure 8(b) in which Veh_s is sending data to Veh_d through Veh_3. As explained before, Veh_s uses fuzzy system for evaluating the links between itself and neighbors. Also it calculates *q-values* for each action (choosing a neighbor for forwarding), before sending any data. Now it may be the case that Veh_s discovers the link (Veh_s-Veh_3) is becoming weak for some reasons (e.g. obstacles, Veh_2 is getting away from Veh_s, and etc.) and at the moment it may become informed about existing a new path toward Veh_d through Veh_2 by receiving *v-value* ($V_{2,d}$) from Veh_2. Then Veh_s may change its path toward Veh_d by calculating *q-value*($Q_d(F,veh_2)$), applying equation (8), then, selecting Veh_2 as the next step. This way, Veh_s can prevent disturbance in sending data toward Veh_d and retransmission requirements. Thus, in this scenario



(a)Veh_s is sending data via Veh_3



(b)Veh_s changes its sending data path

Figure 8. A sample scenario of proposed routing protocol adaptability.

we can see that the proposed protocol can adapt itself to dynamic changes by applying a dynamic model (fuzzy system) and learning new events (the emergence of a new path).

VI. PERFORMANCE EVALUATION

To evaluate the proposed protocol, we have implemented it in Omnet++5.0[43]. We have also used Sumo[44] for generating an actual simulated movement model. The map of a part of Tehran is simulated in Sumo for movements of vehicles with actual traffic rules (traffic lights and signs). The velocity of vehicles is set between 0 to 80 km/h. At each simulation run, the source and the destination of messages are selected randomly, and the number of source/destination pairs is assumed to be between ten and fifteen. The simulation duration is 450s. A 10-15MB file is generated to be sent from a source to a destination. This file can be a video of an event in the city that should be distributed between drivers. A brief of simulation parameters is given in Table 3.

TABLE III. SIMULATION PARAMETERS

Parameter	Value
Simulation Area	2.5km*2km
Mobility model	TraciMobility
Mac Layer	IEEE 8011p
Simulation Duration	450s
Size of Messages	5-10 MB
Communication Range	450m
Number of Runs	20
Data Rate	2Mb/s
Learning Parameters	τ =3, γ=0.3, α=β=0.5

A snapshot of the simulation area in Omnet++ is also deposited in figure 9. Simulation is conducted in two parts. In section 6.1, we have compared the proposed model-based RL algorithm with a model-free RL, and in section 6.2, the proposed RRPV is compared with other protocols namely Geographic Source Routing (GSR)[45], Q-learning Grid based Routing (QGrid)[33], and Q-learning AODV (QLAODV)[46]. Reported results are the average of 20 runs for each simulation. It is assumed that a destination is reachable via multi-hop routing. Whenever the density of the network becomes very low and nodes can't find a neighbor for forwarding packets, the *store-carry-forward* mechanism is used in which packets are carried until a node is found or the timer of carrying is expired. We have utilized the following metrics for comparison:

- Average of Packet Delivery Ratio (PDR): This metric shows the average proportion of the number of packets that are successfully received in the destination to the number of packets sent from the source.

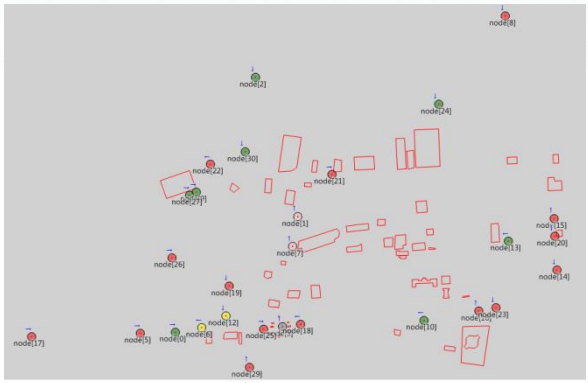


Figure 9. A snapshot of simulation area in Omnet++.

- Average transmission delay: This metric indicates the average duration from when a packet is generated at the source until it is delivered to the destination.
- Message overhead: This metric expresses the average number of control packets needed for routing and delivering a message from the source to the destination

.

A. *Model-Based VS Model-Free*

In this section, we have investigated the effect of using our model-based RL algorithm vs. a model-free RL scheme like Q-learning[35]. In the following charts, the comparison is between our model-based RL algorithm, a model-free, and the optimal mode that is obtained by value-iteration[40] scheme. As mentioned before, high dynamicity is a key feature of VANET, thus having a dynamic model which continually evaluates links, can strongly improve the performance of the routing algorithm. As it was depicted in figure 10 and figure 11, applying the model-based RL could improve PDR and Delay metric, and its operation is also closer to optimal mode. At first, when no routing information is available (i.e. Q-table is empty), both approaches have the same performance, but as time goes, and *v-values* are updated in routing tables, the model-based approach works significantly better, and find more suitable paths, which lead to a higher PDR. In terms of transmission delay, since both approaches use broadcasting for path discovery at the beginning, the transmission delay is substantially high. However, as time elapses and routing information is formed in tables, transmission delay decreases in both methods. As it can be seen in figure 10, the rate of decrease in the model-based approach is much higher than that of the model-free one because the model-based approach can find more stable paths, and as a result, it experiences much lower link breakage in paths.

We have also investigated the capability of the two approaches in dealing with network dynamicity. To this end, we have increased the velocity of vehicles within the range [10, 80] Km/h. As it is depicted in figure 12 and figure 13, increasing the velocity of

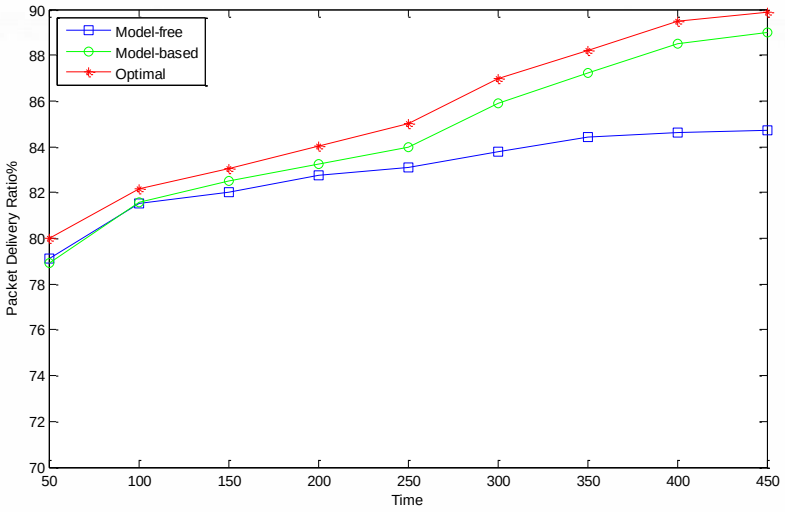


Figure 10. Packet delivery ratio.

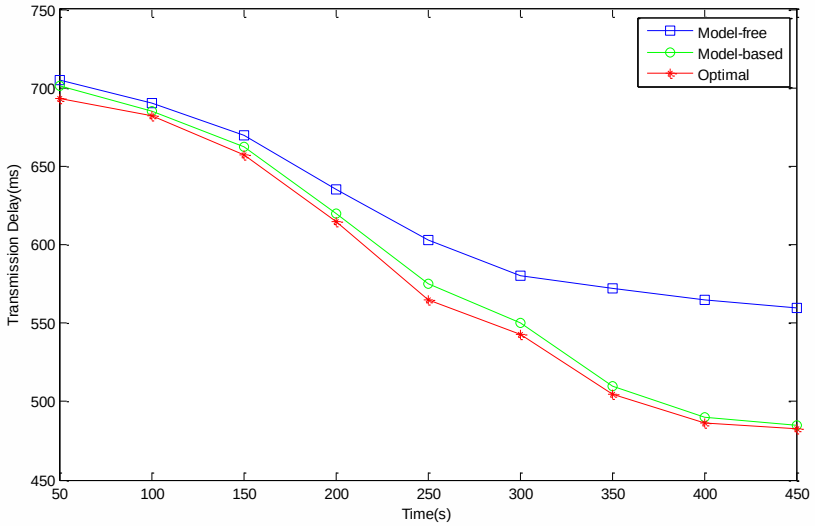


Figure 11. Transmission delay.

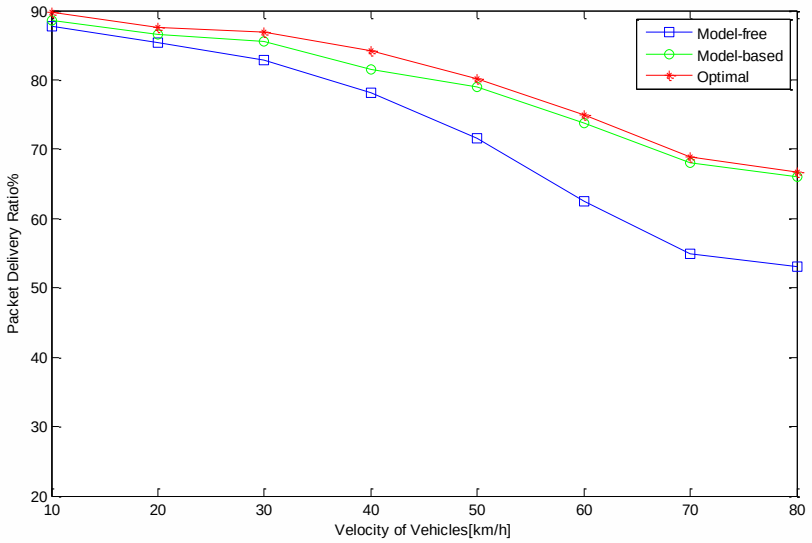


Figure 12. Packet delivery ratio.

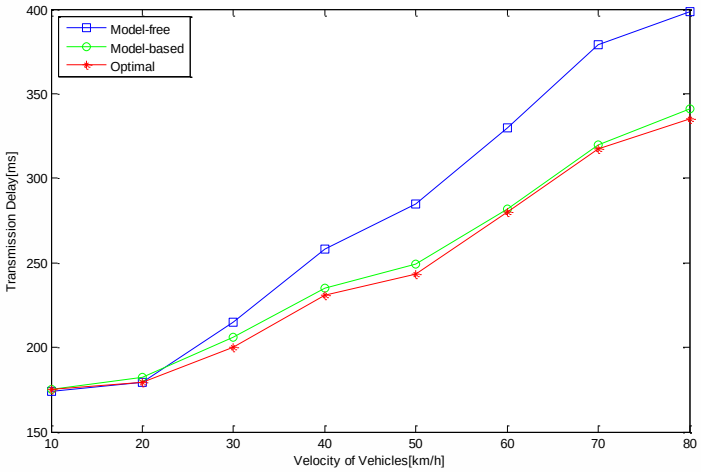


Figure 13. Transmission delay.

vehicles seriously degrades the performance of both model-based and model-free approaches in terms of packet delivery ratio and transmission delay. However, the model-based approach can better adapt itself to such highly dynamic environments, and its performance is less affected in comparison to the model-free approach. This is also due to the fact that the model-based approach is able to find more stable paths than the model-free approach.

B. Comparision between PRPV and Other Protocols

Here we have compared the performance of the proposed protocol with the following protocols:

QGrid[33]: A hierarchical routing protocol that divides the geographic area of the vehicle into grids and then it applies Q-learning to find optimal gird and vehicle toward the destination.

QLAODV[46]: A distributed reinforcement learning routing protocol that works over existing AODV. It uses a Q-learning algorithm to infer network state information and uses a unicast control packet to check the path availability in a real time manner.

GSR[45]: A position-based routing protocol that uses a reactive location service to learn the position of source and destination. It also uses topological knowledge to compute a sequence of junctions that the packet has to traverse.

To this end, we have conducted two experiments; at the first experiment, we have compared the performance of the RRPV by changing network density. The second experiment is devoted to the evaluation of the effect of the velocity of the node on the performance of the proposed routing protocol.

1) Impact of density

In this experiment, we have investigated the impact of the network density on the performance of the proposed algorithm. To this end, we have increased the number of vehicles from 100 to 500. As shown in figure 14, by increasing the network density, the value of PDR is significantly improved. We can see that as the network density increases, the performance of the proposed RRPV also increases in comparison to GSR, QGrid, and QLAODV protocols. This is due to the fact that increasing the number of vehicles will result in more possible links between nodes, and consequently, each node has more chances to learn about stable links.

Figure 15 shows the overhead of routing protocols in comparison to each other. As we can see, the value of this metric increases as the density increases. This is quite natural because increasing the number of nodes results in issuing more routing control messages. As it is illustrated in this figure, the RRPV has a lower overhead in comparison to other protocols. This could be due to the following three reasons: 1) most of the time, RREP(route reply) and RREQ(route request) messages, as the main sources of overhead, are released in the early stages of the network. The number of these messages can be reduced by forming routing Tables. 2) In RRPV, control packets are piggybacked within data packets. 3)The proposed protocol has the capability of learning the dynamics of the network and using more stable links in forming a path between the source and the destination. So, the need for rerouting due to the breakage of formed paths will be substantially decreased in RRPV.

Figure 16 shows the average transmission delay for each of the experimented protocols. Since in RRPV, routing tables are always updated by getting advertisements, each node has the chance to form more suitable paths towards the destination, and thus, packets are delivered in shorter time frames. However, by decreasing network density, this chance decreases, and the average of the delay increases. This is due to the usage of the *store-carry-forward* scheme in RRPV, which ensures a higher packet delivery ratio at the expense of higher delays.

2) Impact of velocity

Velocity is one of the most influencing factors on the performance of routing protocols in VANET. In this section, we have investigated the impact of velocity by changing the maximum velocity of each vehicle within the range [10, 80] Km/h. As shown in figure 17, as velocity increases, the PDR decreases,

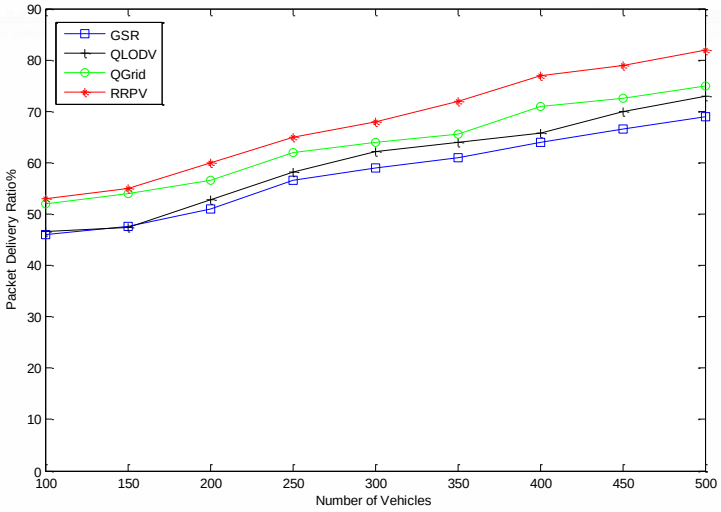


Figure 14. Packet delivery ratio.

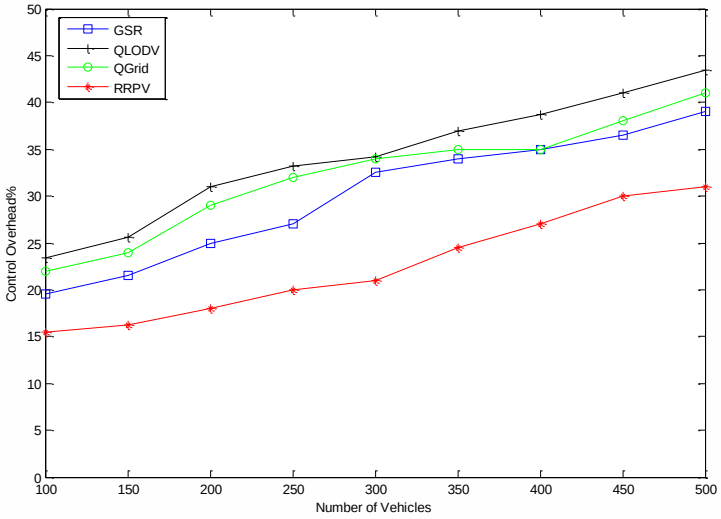


Figure 15. Control overhead ratio.

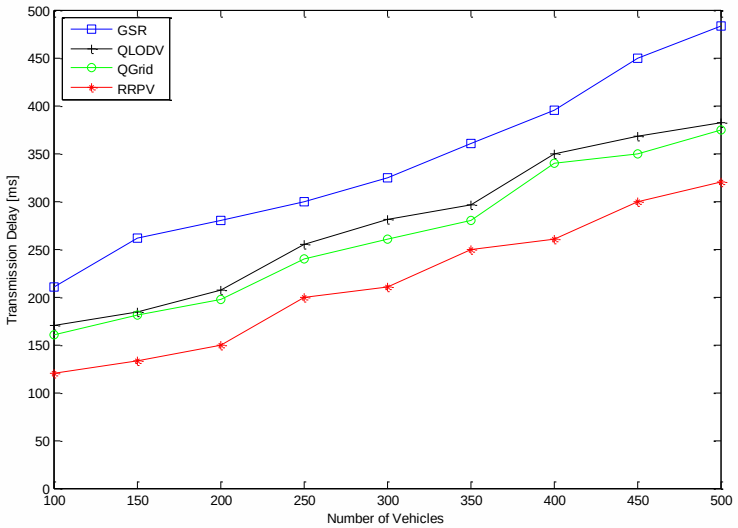


Figure 16. Transmission delay.

because increasing the velocity of each vehicle can lead to the higher dynamic behavior of the network, and consequently a higher number of links in a path are exposed to interruption. In this situation, it is more important to consider the stability factor of links when choosing neighbors.

As it can be seen in figure 17, when the velocities of nodes are low (lower dynamic), the difference between the PDR obtained from RRPV and the two other protocols is negligible. However, as the velocities (higher dynamic) of vehicles increase, this difference becomes more considerable. This can be explained regarding the adaptability of the proposed protocol. The ability of each node in learning via receiving feedback from neighbors (in the form of *v-values*) helps it in choosing better links, and consequently more stable paths.

Figure 18 depicts the impact of vehicles velocity on the packet delay. It is seen that the proposed protocol improves this metric significantly over GSR, QGrid, and QLAODV. One of the main reasons for the increased delay in VANET is the successive breakage of links in a path which will be followed by several retransmissions. This event is severely influenced by the velocity because as the velocity of a node increases, the probability of link breakage is also increased.

According to the creating more stable paths by RRPV, we can see that delay increment in high velocity occurs with a gentler slope in RRPV compared to the two other protocols. On the other hand, by considering the update scheme which is applied in RRPV, most of the time, the routing table has updated routes; thus, different senders can profit from pre-constructed routes, which substantially reduces the delay time of routing and rerouting.

As shown in figure 19, all protocols are influenced by great changes and dynamics of the network in terms of the control message overhead. However, we can see that RRPV has the minimum overhead in comparison to the other two protocols.

One of the main reasons for the increase in the number of control messages is the requirement of maintaining routes and frequent rerouting. In RRPV, by applying a learning technique and using an adaptable scheme in routing, we can reduce this requirement and consequently its overhead.

VII. CONCLUSION

In this paper, we have proposed a new protocol for routing in Vehicular Ad hoc Network (VANET). Our main idea was applying Multi-Agent Reinforcement Learning (MARL) scheme in such a way that nodes can adapt their routing decisions to their environment. We have used model-based Reinforcement learning that is effective for a highly dynamic system. For creating the required state transition model in Multi Agent Reinforcement Learning (MARL), we have used a Fuzzy Logic (FL) system which operates based on different parameters of the links between nodes and their neighbors. The feedback received from an environment is a key component in reinforcement learning; thus, we have used both positive and negative ones. We have performed extensive simulations to evaluate the performance of the proposed protocol and compared it with other protocols, namely Geographic Source Routing (GSR), Greedy Traffic-Aware Routing (QGrid), and Q-learning AODV(QLAODV). We have considered two effective parameters of a network for evaluating the performance of the proposed protocol: velocity and density. The obtained results have shown a considerable improvement in routing metrics, including packet delivery ratio, delay, and overhead. Improvements are resulted from more stable paths, especially when changes and dynamics of the network topology are significantly great.

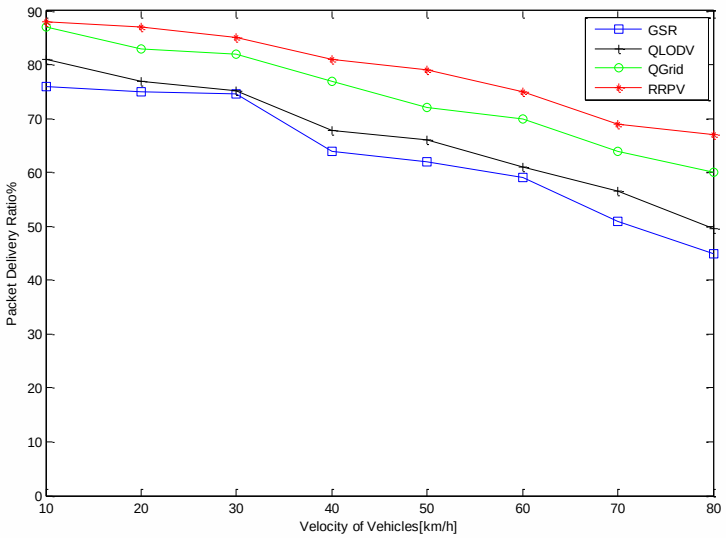


Figure 17. Packet delivery ratio.

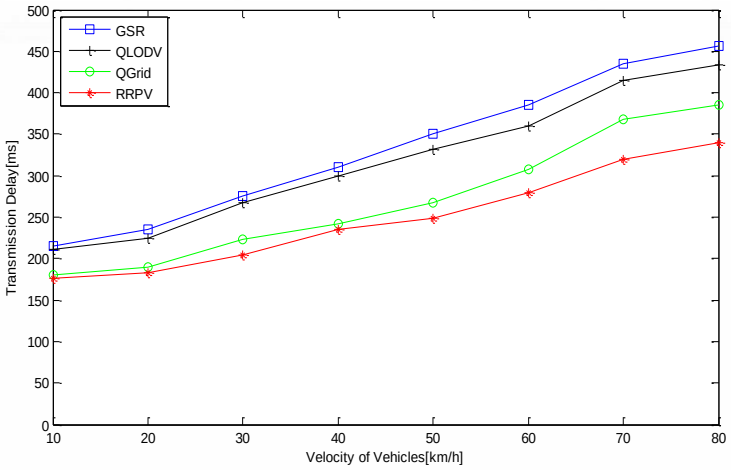


Figure 18. Transmission delay.

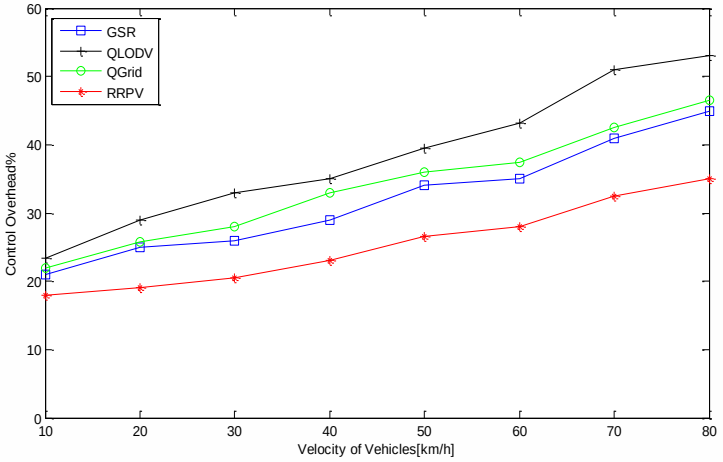


Figure 19. Control overhead ratio.

REFERENCES

[1] Tong, W., Hussain, A., Bo, W. X., & Maharjan, S. (2019). Artificial Intelligence for Vehicle-to-Everything: A Survey. *IEEE Access*, *7*, 10823-10843, doi:10.1109/ACCESS.2019.2891073.

[2] Nazib, R. A., & Moh, S. (2021). Reinforcement Learning-Based Routing Protocols for Vehicular Ad Hoc Networks: A Comparative Survey. *IEEE Access*, *9*, 27552-27587, doi:10.1109/ACCESS.2021.3058388.

[3] Srivastava, A., Prakash, A., & Tripathi, R. (2020). Location based routing protocols in VANET: Issues and existing solutions. *Vehicular Communications*, *23*, 100231, doi:https://doi.org/10.1016/j.vehcom.2020.100231.

[4] Awang, A., Husain, K., Kamel, N., & Aïssa, S. (2017). Routing in Vehicular Ad-hoc Networks: A Survey on Single- and Cross-Layer Design Techniques, and Perspectives. *IEEE Access*, *5*, 9497-9517, doi:10.1109/ACCESS.2017.2692240.

[5] Gao, H., Liu, C., Li, Y., & Yang, X. (2021). V2VR: Reliable Hybrid-Network-Oriented V2V Data Transmission and Routing Considering RSUs and Connectivity Probability. *IEEE Transactions on Intelligent Transportation Systems*, *22*(6), 3533-3546, doi:10.1109/TITS.2020.2983835.

[6] Liu, J., Wan, J., Wang, Q., Deng, P., Zhou, K., & Qiao, Y. (2016). A survey on position-based routing for vehicular ad hoc networks. *Telecommunication Systems*, *62*(1), 15-30, doi:10.1007/s11235-015-9979-7.

[7] Kaur, R., & Rana, D. S. B. (June-2015). OVERVIEW ON ROUTING PROTOCOLS IN VANET. *International Research Journal of Engineering and Technology*, *02*(03), 1333-1337.

[8] Katsaros, K., Dianati, M., Tafazolli, R., & Kernchen, R. CLWPR — A novel cross-layer optimized position based routing protocol for VANETs. In *2011 IEEE Vehicular Networking Conference (VNC), 14-16 Nov. 2011 2011* (pp. 139-146). doi:10.1109/VNC.2011.6117135.

[9] Chettibi, S., & Chikhi, S. (2010). A Survey of Reinforcement Learning Based Routing Protocols for Mobile Ad-Hoc Networks. In (Vol. 162, pp. 1-13).

[10] Mohandas, G., Silas, S., & Sam, S. Survey on routing protocols on mobile adhoc networks. In *2013 International Mutli-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), 22-23 March 2013 2013* (pp. 514-517). doi:10.1109/iMac4s.2013.6526467.

[11] Wang, W., Xie, F., & Chatterjee, M. (2009). Small-Scale and Large-Scale Routing in Vehicular Ad Hoc Networks. *IEEE Transactions on Vehicular Technology*, *58*(9), 5200-5213, doi:10.1109/TVT.2009.2025652.

[12] Perkins, C. E., & Royer, E. M. Ad-hoc on-demand distance vector routing. In *Proceedings WMCSA'99. Second IEEE Workshop on Mobile Computing Systems and Applications, 25-26 Feb. 1999 1999* (pp. 90-100). doi:10.1109/MCSA.1999.749281.

[13] Li, F., & Wang, Y. (2007). Routing in vehicular ad hoc networks: A survey. *IEEE Vehicular Technology Magazine*, *2*(2), 12-22, doi:10.1109/MVT.2007.912927.
[14] Karagiannis, G., Altintas, O., Ekici, E., Heijenk, G., Jarupan, B., Lin, K., et al. (2011). Vehicular Networking: A Survey and Tutorial on Requirements, Architectures, Challenges, Standards and Solutions. *IEEE Communications Surveys & Tutorials*, *13*(4), 584-616, doi:10.1109/SURV.2011.061411.00019.
[15] Blum, J. J., Eskandarian, A., & Hoffman, L. J. (2004). Challenges of intervehicle ad hoc networks. *IEEE Transactions on Intelligent Transportation Systems*, *5*(4), 347-351, doi:10.1109/TITS.2004.838218.
[16] Wang, S. Y., Lin, C. C., Hwang, Y. W., Tao, K. C., & Chou, C. L. (2005). *A practical routing protocol for vehicle-formed mobile ad hoc networks on the roads.* Paper presented at the 2005 IEEE Intelligent Transportation Systems Conference, Oct. 2005
[17] Karp, B., & Kung, H. T. (2000). *GPSR: greedy perimeter stateless routing for wireless networks.* Paper presented at the Proceedings of the 6th annual international conference on Mobile computing and networking, Boston, Massachusetts, USA,
[18] Salkuyeh, M. A., & Abolhassani, B. (2016). An Adaptive Multipath Geographic Routing for Video Transmission in Urban VANETs. *IEEE Transactions on Intelligent Transportation Systems*, *17*(10), 2822-2831, doi:10.1109/TITS.2016.2529178.
[19] Huang, C., & Lin, S. (2014). Timer-based greedy forwarding algorithm in vehicular ad hoc networks. *IET Intelligent Transport Systems*, *8*(4), 333-344, doi:10.1049/iet-its.2013.0014.
[20] Abuashour, A., & Kadoch, M. (2017). Performance Improvement of Cluster-Based Routing Protocol in VANET. *IEEE Access*, *5*, 15354-15371, doi:10.1109/ACCESS.2017.2733380.
[21] Bitam, S., Mellouk, A., & Zeadally, S. (2015). Bio-Inspired Routing Algorithms Survey for Vehicular Ad Hoc Networks. *IEEE Communications Surveys & Tutorials*, *17*(2), 843-867, doi:10.1109/COMST.2014.2371828.
[22] Bi, X., Gao, D., & Yang, M. A Reinforcement Learning-Based Routing Protocol for Clustered EV-VANET. In *2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC), 12-14 June 2020 2020* (pp. 1769-1773). doi:10.1109/ITOEC49072.2020.9141805.
[23] Nahar, A., & Das, D. Adaptive Reinforcement Routing in Software Defined Vehicular Networks. In *2020 International Wireless Communications and Mobile Computing (IWCMC), 15-19 June 2020 2020* (pp. 2118-2123). doi:10.1109/IWCMC48107.2020.9148237.
[24] Roh, B.-S., Han, M.-H., Ham, J.-H., & Kim, K.-I. (2020). Q-LBR: Q-learning Based Load Balancing Routing for UAV-assisted VANET. *Sensors (Basel, Switzerland)*, *20*(19), 5685, doi:10.3390/s20195685.
[25] Saravanan, M., & Ganeshkumar, P. (2020). Routing using reinforcement learning in vehicular ad hoc networks. *Computational Intelligence*, *36*, doi:10.1111/coin.12261.
[26] Wu, J., Fang, M., Li, H., & Li, X. (2020). RSU-Assisted Traffic-Aware Routing Based on Reinforcement Learning for Urban Vanets. *IEEE Access*, *8*, 5733-5748, doi:10.1109/ACCESS.2020.2963850.
[27] Eiza, M. H., Owens, T., Ni, Q., & Shi, Q. (2015). Situation-Aware QoS Routing Algorithm for Vehicular Ad Hoc Networks. *IEEE Transactions on Vehicular Technology*, *64*(12), 5520-5535, doi:10.1109/TVT.2015.2485305.
[28] Li, G., Boukhatem, L., & Wu, J. (2017). Adaptive Quality-of-Service-Based Routing for Vehicular Ad Hoc Networks With Ant Colony Optimization. *IEEE Transactions on Vehicular Technology*, *66*(4), 3249-3264, doi:10.1109/TVT.2016.2586382.
[29] Wu, C., Ji, Y., Liu, F., Ohzahata, S., & Kato, T. (2015). Toward Practical and Intelligent Routing in Vehicular Ad Hoc Networks. *IEEE Transactions on Vehicular Technology*, *64*(12), 5503-5519, doi:10.1109/TVT.2015.2481464.
[30] Zhang, X., Zhang, X., & Gu, C. (2017). A micro-artificial bee colony based multicast routing in vehicular ad hoc networks. *Ad Hoc Networks*, *58*, 213-221, doi:https://doi.org/10.1016/j.adhoc.2016.06.009.
[31] Yang, X., Zhang, W., Lu, H., & Zhao, L. (2020). V2V Routing in VANET Based on Heuristic Q-Learning. *INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL; Vol 15 No 5 (2020): International Journal of Computers Communications & Control (October)DO - 10.15837/ijccc.2020.5.3928*.
[32] Yao, L., Wang, J., Wang, X., Chen, A., & Wang, Y. (2018). V2X Routing in a VANET Based on the Hidden Markov Model. *IEEE Transactions on Intelligent Transportation Systems*, *19*(3), 889-899, doi:10.1109/TITS.2017.2706756.
[33] Li, F., Song, X., Chen, H., Li, X., & Wang, Y. (2019). Hierarchical Routing for Vehicular Ad Hoc Networks via Reinforcement Learning. *IEEE Transactions on Vehicular Technology*, *68*(2), 1852-1865, doi:10.1109/TVT.2018.2887282.
[34] Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1996). Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research*, *4*, 237–285.
[35] C.Watkins (1989). *Learning from delayed rewards*. Student thesis, dissertation,King's College, Cambridge, U.K.
[36] Dorigo, M., & Caro, G. D. Ant colony optimization: a new meta-heuristic. In *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406), 6-9 July 1999 1999* (Vol. 2, pp. 1470-1477 Vol. 1472). doi:10.1109/CEC.1999.782657.
[37] Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, *8*(3), 338-353, doi:https://doi.org/10.1016/S0019-9958(65)90241-X.
[38] Cintula, P., Fermüller, Christian G. and Noguera, Carles (2017). Fuzzy Logic. *Stanford Encyclopedia of Philosophy*.
[39] Mamdani, E. H. a. S. A. (1975). An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, *7*(1), 1-13.
[40] Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction*: MIT Press.
[41] Siddique, M. (2009). *Fuzzy Decision Making Using Max-Min Method and Minimization Of Regret Method(MMR)* Blekinge Institute of Technology
[42] Bellman, R. (1957). *Dynamic Programming. Princeton, NJ:* (Vol. Press): Princeton Univ.
[43] OMNeT++ Community, OMNeT++ Network Simulator, .
[44] SUMO Simulation of Urban Mobility. [Online].
[45] Lochert, C., Hartenstein, H., Tian, J., Fussler, H., Hermann, D., & Mauve, M. A routing strategy for vehicular ad hoc networks in city environments. In *IEEE IV2003 Intelligent Vehicles Symposium. Proceedings (Cat. No.03TH8683), 9-11 June 2003 2003* (pp. 156-161). doi:10.1109/IVS.2003.1212901.
[46] Wu, C., Kumekawa, K., & Kato, T. (2010). Distributed Reinforcement Learning Approach for Vehicular Ad Hoc Networks. *IEICE Transactions*, *93-B*, 1431-1442, doi:10.1587/transcom.E93.B.1431.

Omid Jafarzadeh received the B.Sc. in Computer Engineering from the Islamic Azad University of Sari and the M.Sc. degree in Computer Engineering from Islamic Azad University of Qazvin. Currently he is Ph.D. student in Computer Engineering, Islamic Azad University of Qazvin. He has teaching experience in Network for more than 10 years in

Islamic Azad University of Behshahr. His research interests are mainly on Wireless Ad hoc Networks.

Mehdi Dehghan received his B.Sc. degree in Computer Engineering from Iran University of Science and Technology (IUST), Tehran, Iran in 1992, and his M.Sc. and Ph.D. degrees from Amirkabir University of Technology (AUT), Tehran, Iran in 1995, and 2001, respectively. He joined the Computer Engineering and Information Technology (CEIT) Department of Amirkabir University of Technology in 2004. Currently, as a professor, he is the director of the Mobile Ad hoc and Wireless Sensor Lab at AUT. His research interests include High Speed Networks, Network Management, Mobile Ad hoc Networks and Fault-Tolerant Computing Advanced XED (AXED) Algorithm.

Hadi Sargolzaey received the B.Sc. in Electronic Engineering and the M.Sc. degree in Digital Electronic Engineering from Ferdowsi University of Mashhad and Sharif University respectively. He also received his Ph.D. degree in Communication and Network Engineering from University Putra Malaysia. He is currently an Assistant Professor with Faculty of Computer Engineering in Islamic Azad University of Qazvin and has teaching experience in Data Communication Network for more than 20 years. His research interests are mainly on Wireless Data Communication Networks and Microprocessor Systems.

Mehdi Esnaashari received the B.Sc., M.Sc., and the Ph.D. degrees in computer engineering, from the Amirkabir University of Technology, Tehran, Iran, in 2002, 2005, and 2011, respectively. Prior to his current position, he was an Assistant Professor with Iran Telecommunications Research Center, Tehran. He is currently an Assistant Professor with faculty of Computer Engineering, K. N. Toosi University of Technology, Tehran, Iran. His current research interests include Computer Networks, Learning Systems, Soft Computing, and Information Retrieval.