Learning for Multiple-Relay Selection in a Vehicular Delay Tolerant Network

YUZHEN DONG[1], **FUQUAN ZHANG**[1], **INWHEE JOE**[2], **(Member, IEEE),**
HAIFENG LIN[1], **WANGUO JIAO**[1], **AND YAN ZHANG**[1]
[1]College of Information Science and Technology, Nanjing Forestry University, Nanjing 210000, China
[2]Department of Electronics and Computer Engineering, Hanyang University, Seoul 04763, South Korea

Corresponding author: Fuquan Zhang (zfq@njfu.edu.cn)

ABSTRACT In a vehicular delay tolerant network (VDTN), there is no static connection, and the network behavior is highly temporal. This makes determining the routing protocol critical for the performance of a network. However, traditional routing technology rarely considers the influence of a VDTN's selfish nodes. Under ideal conditions, all nodes will try to store and transfer as many messages as possible. However, selfish nodes may not transfer messages to other nodes due to limited resources. Taking selfish nodes into account is necessary for improving the performance of a VDTN. In this paper, we calculate each node's credit value by recording each node's behavior when messages are transferring to avoid selecting selfish nodes as the relay. We present an efficient VDTN specific multi-copy routing algorithm that combines Q-learning and the credit value to determine whether a candidate node is suitable for delivering a message. Because multi-copy routing protocols require a high buffer to store messages, we also optimize the node buffer to reduce network congestion. The proposed algorithm is evaluated by a number of different performance metrics, such as delivery probability, network overhead, and message latency. The proposed algorithm achieves better results in different configurations and provides improved delivery probability, low message latency, and network overhead.

INDEX TERMS Delay-tolerant networks, multi-copy, routing, Q-learning, buffer management.

I. INTRODUCTION

Vehicle-to-vehicle communication is necessary for different applications in the field of intelligent vehicle systems, such as adaptive content delivery services, automatic emergency service scheduling, intelligent traffic notification, and routing [1]–[4]. In these applications, reliable communication between vehicles is the common challenge.

It is important to ensure that a message is delivered from a source to its intended destination in a reasonable time. Due to the lack of direct point-to-point connections, routing protocols have an important effect on network performance [5]–[8]. Capturing, storing, and forwarding messages are typical operations for nodes in a vehicular delay tolerant network (VDTN). However, a node has a limited ability to send and receive messages. These include limitations of communication range, message life, and memory and

The associate editor coordinating the review of this manuscript and approving it for publication was Yassine Maleh.

computing power, which all contribute to a lack of network availability.

In addition, as nodes continue to move, the routing protocol must ignore and be independent of the constantly changing network topology. In practice, every node has to store messages in its message buffer until they have met other nodes [8], [9]. However, when traffic conditions are crowded, it is hard to ensure that one node keeps the message in a limited buffer until the destination node is encountered. In reality, the message has a time to live (TTL) value at the end. TTL requires a node to transfer messages to other nodes before the TTL expires. A method to overcome this limitation is to copy the message to multi-relay nodes (that is, to generate several copies of the message) so that the probability of encountering the intended destination node of the message in a multi-copy protocol is increased.

In a VDTN system, some nodes will refuse to transfer messages from other nodes due to insufficient resources. Such nodes are termed selfish, and the selfish behavior of the nodes

may affect the performance of the entire route [10], [11]. The credit value is used to evaluate whether a node is selfish or not. It is important to stimulate selfish nodes to cooperate in transferring messages. When two nodes are connecting, each node calculates the comprehensive credit value of the other nodes. A node with a credit value below the credit threshold is considered to be a selfish node, and normally, the threshold value is set as 0.3 [11]. At this point, no nodes transfer messages from selfish nodes. In order to improve its credit value, a selfish node has to relay messages so that other unselfish nodes will transfer its messages.

This paper presents an efficient VDTN multi-copy routing algorithm that uses Q-learning to determine if a candidate node is suitable for transferring a message based on its credit value. In addition, each node keeps a successfully transferred message list to store the information of a successfully transferred message. Nodes can delete already delivered messages by exchanging the list when they meet each other. This buffer optimization method was integrated into the routing algorithm to alleviate the occurrence of network congestion. Simulation studies have shown that our method is more effective than previous methods. Our work makes the following contributions.

1) We propose a new method called Q-learning multi-copy routing (QMCR) that combines the Q-learning algorithm and the credit value to select the candidate node.
2) The proposed algorithm has efficient buffer management that could alleviate the intrinsic buffer requirement of multi-copy routing in VDTN.
3) We conduct a detailed simulation of the proposed algorithm, and the results show that it has good performance.

II. RELATED WORK

Multi-copy routing protocols generate multiple message copies and transfer them. Message copies increase the possibility of delivery, and the message delivery rate is higher than single-copy routing protocols. However, at the same time, multi-copy routing protocols also consume too much of the node cache and bandwidth because the network is flooded with a large number of message copies. As a consequence, the probability of network congestion is greater [12]–[14]. The epidemic router based protocol (ER) is a typical multi-copy routing protocol that will transfer as many messages as possible [12], [15], [16]. In order to alleviate this type of problem, The spray and wait based protocol (SAW) limits the number of message copies [13]. The PRoPHET router based protocol (PR) calculates the possibility of transmission based on the history information between nodes [17]. ER, SAW, and PR are the conventional multi-copy routing protocols. They are widely used as benchmarks to evaluate the performance of improved routing protocols in VDTN [12]–[17].

As selfish nodes may affect the performance of a VDTN, a selfish DTN message forwarding mechanism based on node behavior analysis (BNBA) was proposed [18]. It studied the cooperation and noncooperation behaviors between the encountered nodes. A model of the node state probability transformation is established and the process of message delivery between nodes is predicted.

To avoid selfish nodes being unwilling to cooperate due to selfish behavior or to save their own resources from being compromised, a reputation system with different mechanisms (allowing punishment of selfish nodes in different ways) was proposed [19]. It adapted the reputation system to perform together with a hybrid system with the main goal of incentivizing selfish nodes to share their resources with others instead of immediately excluding them from the network. In order to alleviate the negative impact of selfish nodes on networks, the reputation based spray protocol (RBS) records a node's behavior when messages are transferring [11]. It combines selfish behavior and inability behavior while revealing very little private information for protocol practice. The protocol focuses on private information to make RBS widely accepted and applied in practice.

The behavior of selfish nodes may waste much buffer. Therefore, it is important to take selfish nodes into account when selecting the relay node. Table 1 summarizes the main technical features of these related routing protocols.

Currently, Q-learning is widely used in machine learning where an agent is able to take actions by learning from the previous conditions of a system [21]–[25]. Q-learning has been used to automatically design efficient network architectures [26], [27]. The target of Q-learning is learning a method and telling an agent what action should by taken according to its state. It does not need an environmental model and can handle the problems of random conversion and rewards [21]–[23]. For any finite Markov decision process, Q-learning finds an optimal strategy, that is, starting from the current state, it maximizes the expected value of the total reward in any and all subsequent steps. Given the infinite exploration time and partial random strategy, Q-learning can determine the best action selection strategy for any given finite Markov decision process. The "Q" named function returns a reward value, namely, the "quality" of the action taken in a given state [24]. However, integration of the credit of nodes, Q-learning, multi-copy, and buffer management have not been well considered or utilized in the design of the protocol of VDTN. Therefore, exploration of the integration of such factors in the development of a protocol is essential.

This paper presents an efficient VDTN multi-copy routing algorithm that uses Q-learning to determine if a candidate node is suitable for transferring a message based on an evaluation of the node's selfishness.

III. PROPOSED PROTOCOL
A. SYSTEM MODEL
In this paper, the scenario we consider is when vehicle (nodes) are randomly scattered around a map and moved, which is shown in Fig. 1. Each node has a fixed communication range.

TABLE 1. Summary of related routing protocols.

Reference	Node selfishness	Multi-copy	Buffer optimization	Technique
[11]	✓	✓	✗	Reputation based spray protocol
[12]	✗	✓	✗	Adaptive probabilistic infection and limited-time forwarding mechanisms
[13]	✗	✓	✗	Quality of node (QoN)
[14]	✗	✓	✗	K-anycast communication
[15]	✗	✓	✓	Broadcasting Acknowledgements
[16]	✗	✓	✓	Replication based and network coding (NC) based epidemic routing protocols
[17]	✗	✓	✗	Energy consumption
[18]	✓	✓	✗	Based on node behavior analysis
[19]	✓	✓	✗	Reputation system together with a hybrid system
Proposed method	✓	✓	✓	Q-learning, credit value, and buffer optimization

Communication is connected when the other nodes enter its communication range. The number of nodes in the map is N, and the set of nodes is *SoN*. Communication between the nodes is denoted as $N2NC_i = \{1, 2, \ldots, N\}$, $i \in SoN$. Each N2NC pair has a transmitter (*N2N_T*) and a receiver (*N2N_R*) to relay and receive messages.

Q-learning is a value-based algorithm in reinforcement learning. The value of Q (s, a) is the income that can be realized by taking action a (a ∈ A) in state s (s ∈ S). The expectation is that the system will return a reward R by taking action. Therefore, the key point is to establish a Q-table to store the Q-value, which is affected by the action and state, and to take the action that can maximize the benefit [23]–[25]. Thus, the expectation of the strategy with the largest cumulative reward is shown as (1) [28].

$$\text{goal} : max_\pi E[\sum_{t=0}^{H} \gamma^t R (S_t, A_t, S_{t+1}) |\pi] \qquad (1)$$

We use the Q-learning method to determine if the candidate node is suitable for relaying a message. The learning environment of the Q-learning algorithm is the entire network. Each node is a learning agent. By relaying messages to the nodes it meets, each node can learn from the current network. Q-learning uses the time difference method to perform offline learning and solves the optimal strategy by using the Bellman equation. The Bellman equation is actually the conversion relationship of the value action function [27].

At time t, each node is moving based on the map, and there is no connection. When the connection is up, at time $t+1$, the operation is to choose the available candidate node for message relaying. If the candidate node is suitable, messages will be transferred. Every node keeps a Q-table, where every Q-value represents the activity of a node in the network [29]–[31].

B. CALCULATION OF THE CREDIT VALUE

In the VDTN system, some nodes will refuse to transfer messages from other nodes due to their own insufficient resources. Such nodes are called selfish nodes, and the selfish behavior of the nodes may affect the performance of the entire route. In this paper, the selfishness of a node is described by the credit value, which can be divided into those directly and indirectly obtained from the nodes. The node j calculates the credit value of the node i, R_{ij}, in (2), including two parts: one part is calculated directly from the recorded data of node j, and the other part is indirectly derived from the remark value of the other nodes [11], [28]. R_{ij}^d and R_{ij}^{in} are the direct credit value and indirect credit value. The weight value is α.

$$R_{ij} = \alpha \times R_{ij}^d + (1 - \alpha) \times R_{ij}^{in}, 0 \leq \alpha \qquad (2)$$

The direct credit value is calculated in (3). The message number that node j transfers to node i is recorded as M_s. The messages number that node i is relayed for node j is recorded as M_f. The indirect credit value is calculated in (4). It is the mean of all the other nodes' direct credit values. The algorithm for calculating the credit value is shown in Algorithm 1. The algorithm for calculating the credit value is shown in Algorithm 1.

$$R_{ij}^d = \frac{M_f}{M_s}, \quad 0 \leq R_{ij}^d \leq 1 \qquad (3)$$

$$R_{ij}^{in} = \frac{1}{n - 2} \sum_{k=1}^{n} R_{ij}^d, \quad k \neq i, k \neq j \qquad (4)$$

C. UPDATE Q-VALUE IN THE Q-TABLE

The Q-learning algorithm is shown as Algorithm 2. Firstly, an initial state-action pair is obtained. After the first iteration, from the current state-action pair, we can obtain an immediate
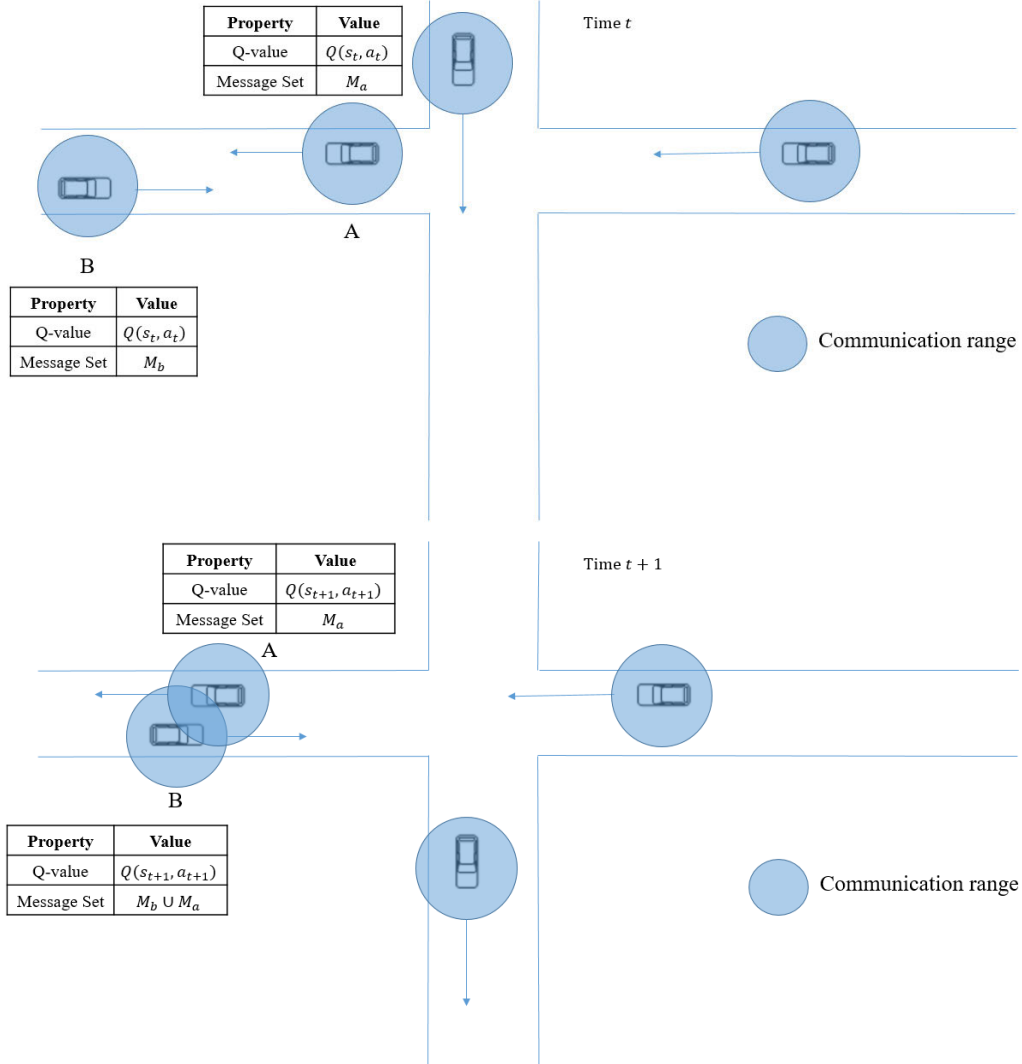
FIGURE 1. System model.

reward and a new state. Then, the algorithm updates the state-action pair as expressed by (5). The optimal policy indicating an action to be taken at each state is maximized for all states.

The update process of the Q-value in the Q-table is shown as (5). The default Q-value in the Q-table is 0.

$$Q_m(s, a) \leftarrow \alpha \times \left[R + \gamma \times max_{y \in N} Q_n(s, a) \right]$$
$$+ (1 - \alpha) \times Q_m(s, a) \quad (5)$$

where γ is the reward decay factor, and α is the learning rate. N is the set of nodes that have met n before. Generally, α is set as 0.3, and γ is set as 0.8 [32]. The maximum Q-value in n's Q-table is $max_{y \in N} Q_n(s, a)$. The reward R is affected by the credit value and the times meet other nodes. The node with a credit value below the credit threshold is considered to be the selfish node, and normally, the threshold value is

set as 0.3 [11]. The algorithm updating the Q-table is shown in Algorithm 3.

Fig. 2 shows the process of updating the Q-value in the Q-table. In Fig. 2(a) and (b), node A and node B have built a connection. If A and B have never met before, and both of their credit values are greater than the threshold, but B has met more nodes than A. After updating the Q-table, the maximum Q-value in A has been updated, and the updated Q-value in B is equal to the maximum Q-value of B. Therefore, they both chose to transfer messages to each other. In Fig. 2(c) and (d), A and C have built a connection. They assume that both of their credit values are greater than the threshold, but C has met more nodes than A. After updating the Q-table, the maximum Q-value in A has been updated, so A choses to transfer messages to C. However, the updated Q-value in C is lower than the maximum Q-value. C chose not to transfer messages to A.
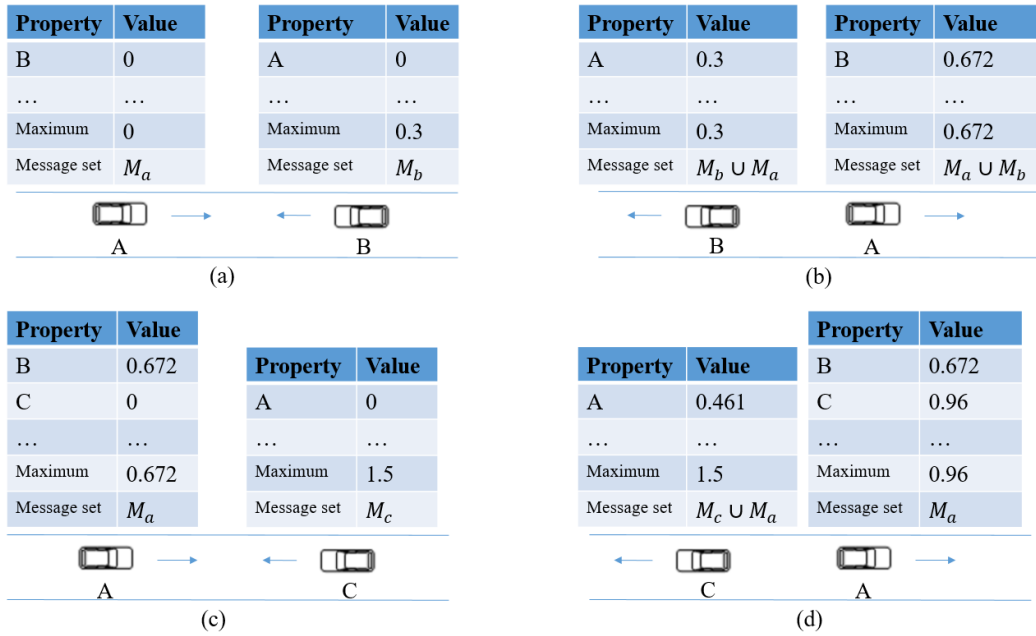
(a)

Property	Value
B	0
...	...
Maximum	0
Message set	M_a

Property	Value
A	0
...	...
Maximum	0.3
Message set	M_b

A → ← B

(b)

Property	Value
A	0.3
...	...
Maximum	0.3
Message set	$M_b \cup M_a$

Property	Value
B	0.672
...	...
Maximum	0.672
Message set	$M_a \cup M_b$

← B   A →

(c)

Property	Value
B	0.672
C	0
...	...
Maximum	0.672
Message set	M_a

Property	Value
A	0
...	...
Maximum	1.5
Message set	M_c

A → ← C

(d)

Property	Value
A	0.461
...	...
Maximum	1.5
Message set	$M_c \cup M_a$

Property	Value
B	0.672
C	0.96
...	...
Maximum	0.96
Message set	M_a

← C   A →

FIGURE 2. Process of updating the Q-table.

Algorithm 1 Algorithm to Calculate the Credit Value

➢ M_f : message number in which node i relayed for node j

➢ M_s : messages number in which node j transfers to node i

➢ R_{ij}^d : direct credit value

➢ R_{ij}^{in} : indirect credit value

➢ R_{ij} : credit value

Initialization;

Connection Up

  If (Transfer message)

    M_f + +;

    M_s + +;

    updateR_{ij}^d();

    updateR_{ij}^{in}();

    updateR_{ij}();

Connection Down

Algorithm 2 Q-Learning Algorithm

Initialize the table $Q(s, a)$ arbitrarily for each state-action pair (s, a);

Observe the current state s;

Begin loop

  From the current state-action pair (s, a), execute action a and obtain the immediate reward R and a new state s';

  Select an action a' based on the state s' and update the table for $Q(s, a)$;

  Replace $s \leftarrow s'$;

End loop

Output: $\pi^* (s) = max_\pi E[\sum_{t=0}^H \gamma^t R (S_t, A_t, S_{t+1}) | \pi]$;

D. NODE BUFFER OPTIMIZATION

Normally, a node tries to transfer messages as much as possible in a multi-copy protocol. Under ideal conditions, this type of protocol can bring a very high delivery probability [15], [33], [34]. However, it requires nodes to have a large buffer size so that they can store as many copies of messages as possible. In fact, the buffer size of each node in a VDTN is limited. This limits the number of messages that can be stored.

A multi-copy protocol can lead to network congestion, and this will seriously affect the delivery probability [16], [35], [36]. As shown in Algorithm 4, to alleviate the network congestion, we integrate a node buffer optimization algorithm. Each node in the VDTN keeps a list of messages that are transferred successfully. In the beginning, this list is null. If a node transfers a message to the target node successfully, this message's information will be recorded in the node's list. When it meets other nodes, it will share its list to others so they can update their lists. Then, they can check and delete the successfully transferred message (already delivered messages) from their buffers to make space.

IV. EVALUATION OF THE PROPOSED ALGORITHM

The opportunistic network environment (ONE) simulator is widely used to simulate different types of environment in a VDTN. We evaluate several multi-copy algorithms and the proposed algorithm under various simulation configurations by using the ONE simulator and report our results here.

TABLE 2. Parameter configurations for simulations.

Parameters	Values
Movement model	Map-based
Buffer size	10 – 50 M
Wait time	0–120s
Maximum number of nodes	350
Maximum speed	20 m/s
Transfer speed	2 Mbps
Transfer range	10 m
Event generators	5-35 s
Message sizes	500 kB–1 MB
Group number	5

Algorithm 3 Algorithm to Update the Q-Table

➢ R: reward value
➢ t_x : time this node has met others
➢ t_c : time the candidate node has met others
➢ c_x : this node's credit value
➢ c_c : candidate node's credit value
Initialization;
Connection Up
  Get candidate node's maximum Q-value;
  Update the credit value;
  Update the times to meet the other nodes;
  If ($t_x \leq t_c \&\&c_c \geq$ threshold)
    $R = 2$;
  Else, if ($t_x > t_c \&\&c_c <$ threshold)
    $R = 1$;
  Else,
    $R = 0$;
  Update the Q-value in the Q-table;
    If (The updated Q-value is the maximum value in the Q-table)
    Message transfer;
  Else,
    Do not transfer message;
Connection Down

Algorithm 4 Algorithm Optimizing the Node Buffer

Initialization;
Connection Up
  If (Other node's list is different)
    updateList();
  for (i = 0;i < list.size;i++)
    Locate and delete messages;
  Connection Down

---

1. For simulating different network loadings, we use a group of different time intervals. Specifically, we change it from 5 s (heavily loaded) to 35 s (lightly loaded).
2. The number of nodes is from 150 (low density) to 350 (high density).
3. We change the speed of the nodes to 5 m/s, 10 m/s, 15 m/s, and 20 m/s.
4. The nodes are running under a map-based movement model.

For each effective configuration, we recorded the performance metrics, including the delivery probability, message latency, network overhead, and average hops. The parameter configurations for simulations are shown in Table 2. The result of five different multi-copy routing protocols (QMCR, RBS, ER, PR, and SAW) in a VDTN is shown in Figs. 3 to 6.

A. RELATIONSHIP BETWEEN VDTN PERFORMANCE AND THE MESSAGE GENERATION INTERVAL

With the increase in the message generation interval, the number of generated messages decreases. Consequently, the pressure of the relay copies decreases. When the message generation interval is long (TTL = 35), all the algorithms realize better performance. From Fig. 3(a) we can learn that the proposed method provides almost 92% delivery probability. This is about 13% better than SAW, which is second and about 56% better than PR, which is the lowest performing. When TTL = 5, the message generation will be frequent and that may lead to a heavily loaded network. However, according to the simulation, the proposed method offers a 46% delivery probability while all other methods expect a SAW offer of about 23% delivery probability. This offers an approximately 23% extra delivery probability.

From Fig. 3(b), the overhead ratio of these algorithms has an upward trend except for SAW. This is because SAW has limited the number of message copies. The proposed method has the second lowest overhead ratio. From Fig. 3(c), the other algorithms do not show an obvious trend in message latency. With the TTL increasing, the proposed method's delivery probability is increasing, while its message latency is decreasing. From Fig. 3(d), we can see that the changes in the average hops are not obvious.
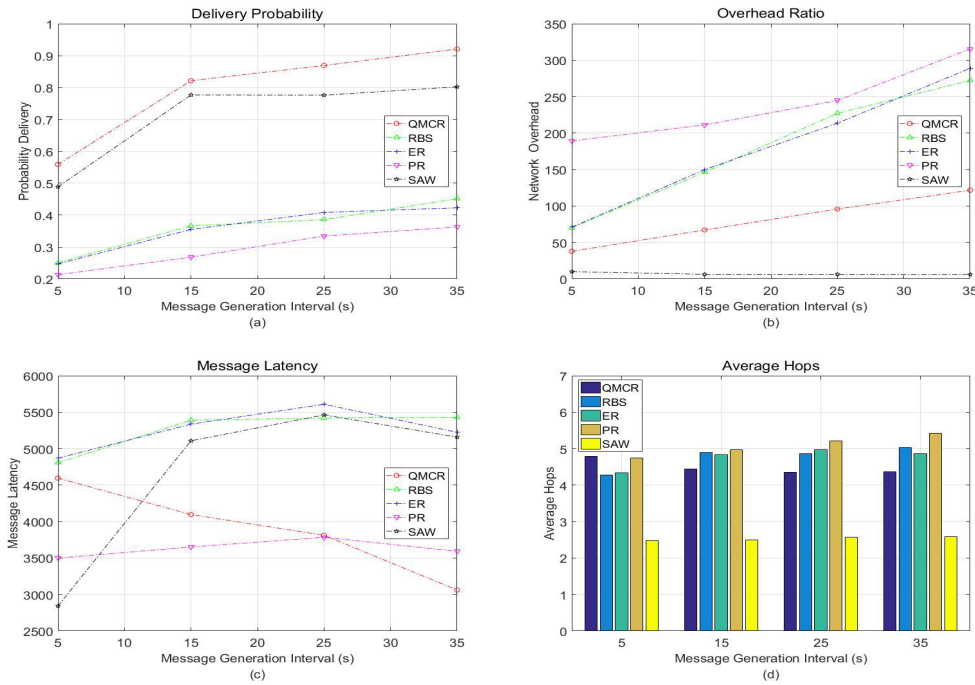
FIGURE 3. Relationship between the VDTN performance and the message generation interval.
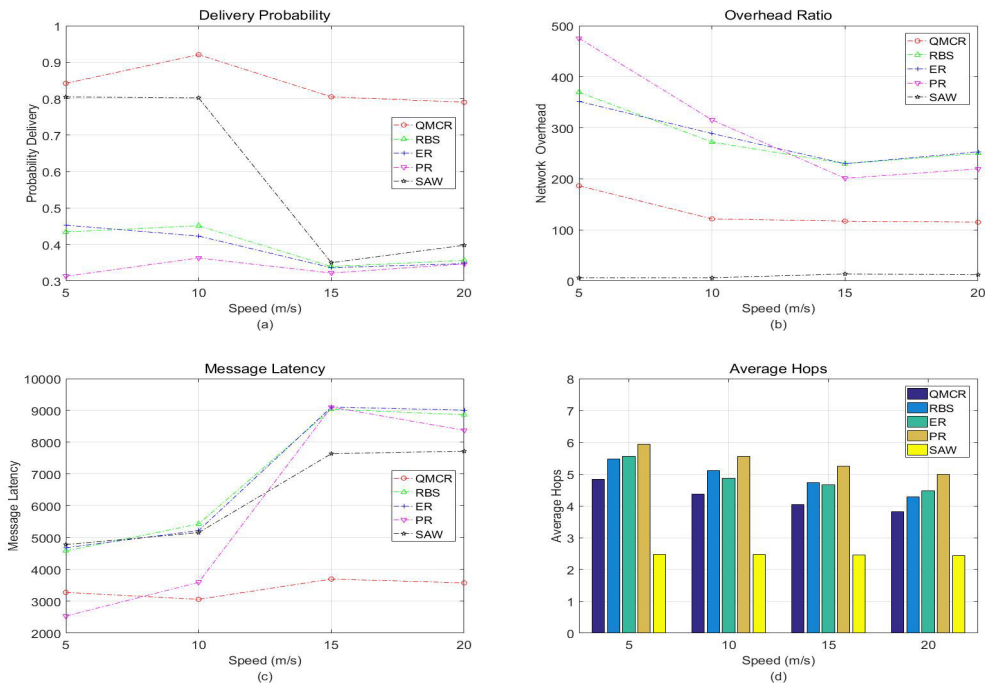

FIGURE 4. Relationship between VDTN performance and speed.

B. RELATIONSHIP BETWEEN THE VDTN PERFORMANCE AND SPEED

While taking the node speed into account, Fig. 4(a) shows that the trend of the proposed method's delivery probability rises first and then falls. This change indicates that the proposed method is suitable at low or medium speed. The lower relative speed means that moving the ability of the nodes is insufficient. The probability of encountering other nodes is relatively low. Vice versa, a faster relative speed leads to more connections, and this may cause more invalid message copies,
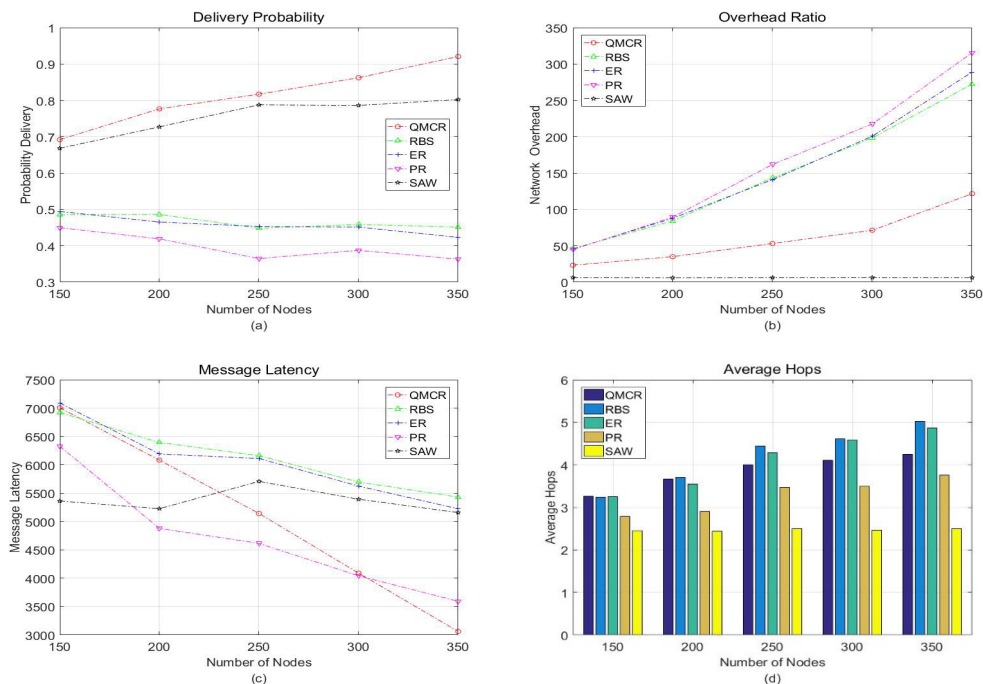
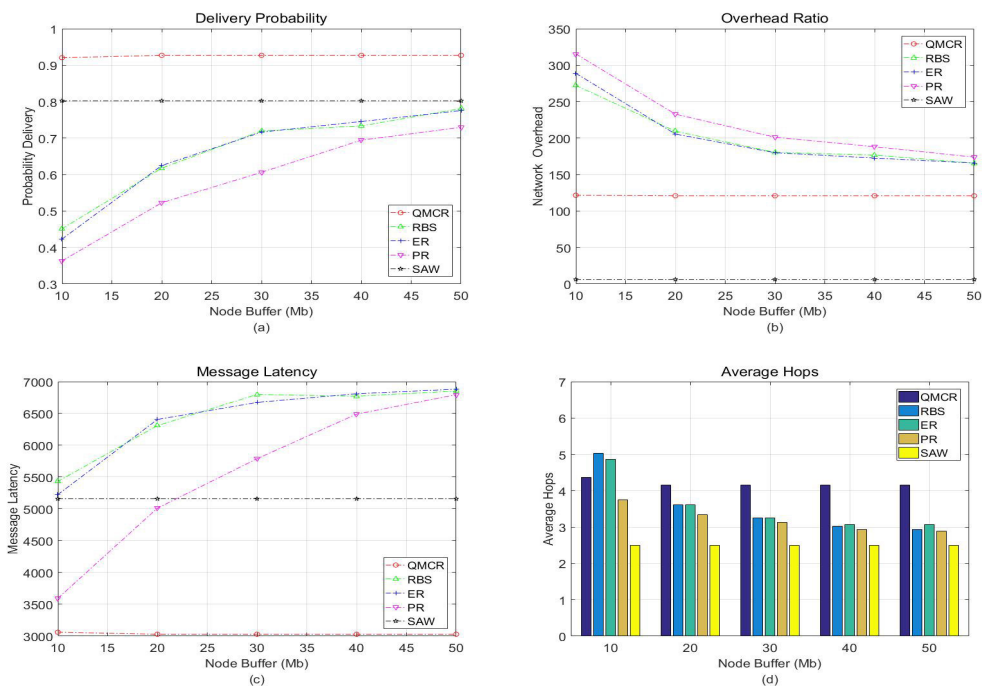**FIGURE 5.** Relationship between VDTN performance and number of nodes.

FIGURE 6. Relationship between VDTN performance and node buffer.

so network congestion is more likely to happen. As indicated in Fig. 4, all the other algorithms start to deteriorate in delivery probability while the node speed is increasing. It is worth mentioning that the delivery probability is affected when the relative speed increases to 15 m/s. The faster relative speed of the nodes may result in unstable connections between the nodes, and this may affect the delivery probability.

From Fig. 4(b) and (c), we can learn that the node speed does not have too much influence on the proposed method's overhead ratio and message latency. From Fig. 4(d), the average hops decrease as the speed increases.

C. RELATIONSHIP BETWEEN VDTN PERFORMANCE AND NUMBER OF NODES

Similar observations can be observed when considering the relationship between VDTN performance and the number of nodes (Fig. 5). Specifically, the results show that the proposed algorithm realizes better performance for any node number. However, this is not the case for other algorithms. While a node's number is increasing, the delivery probability of at least three of the other algorithms deteriorates.

The large number of nodes may result in more available data communication links and that can increase the number of transmitted messages. However, in this case, unlimited message copies may lead to a waste of effort in storing and relaying the copies of messages. From Fig. 5(b), this results in an increased overhead ratio in multi-copy protocols (RBS, ER, and PR). From Fig. 5(c) and (d), more nodes in a network may lead to more communication paths between nodes. This causes lower message latency and higher average hops.

D. RELATIONSHIP BETWEEN VDTN PERFORMANCE AND THE NODE BUFFER

Normally, the node buffer is an important factor that affects VDTN performance in multi-copy protocols. The number of stored messages is decided based on the size of the node buffer. It can be observed from Fig. 6 that when the node buffer begins to increase, the performance of ER, PR, and RBS improves. SAW has a limitation in terms of the number of copies of a message, so it does not require much node buffer. The proposed method (QMCR) has a node buffer optimization algorithm, so the larger node buffer shows less improvement. This result indicates that QMCR may be more suitable for network conditions where the available buffer is insufficient.

V. CONCLUSION AND LIMITATIONS

This paper proposes a multi-copy routing algorithm based on Q-learning. It introduces a message buffer management mechanism to cope with situations where message copies exist in the buffer of nodes. Serval existing algorithms are used to compare with QMCR. Results show that QMCR's overall routing performance (in delivery probability, message latency, and network overhead) is improved. QMCR provides good delivery probability at different message generation intervals, speeds, and the number of nodes (see Figs. 3 to 6). In addition, it also has a low overhead ratio. Although the results of the simulation show that QMCR do not have the lowest latency value, it is still acceptable compared to other algorithms.

Although the proposed algorithm provides relatively good performance, it also has limitations. Since QMCR has node buffer optimization, it does not require a large node buffer. However, with the development of hardware techniques, a node's buffer is becoming larger making buffer optimization no longer necessary in the future.

Despite this limitation, our work has wider applications. Although QMCR is used in the field of VDTN, it can also be easily applied to the Internet of Things [37].

CONFLICTS OF INTEREST

The authors declare that there is no conflict of interest regarding the publication of this paper.

REFERENCES

[1] H. Zhu, K.-V. Yuen, L. Mihaylova, and H. Leung, "Overview of environment perception for intelligent vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 10, pp. 2584–2601, Oct. 2017.

[2] N. Mahajan, A. Hegyi, S. P. Hoogendoorn, and B. van Arem, "Design analysis of a decentralized equilibrium-routing strategy for intelligent vehicles," *Transp. Res. C, Emerg. Technol.*, vol. 103, pp. 308–327, Jun. 2019.

[3] J. Jiang and G. Han, "Routing protocols for unmanned aerial vehicles," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 58–63, Jan. 2018.

[4] A. M. Bozorgi, M. Farasat, and A. Mahmoud, "A time and energy efficient routing algorithm for electric vehicles based on historical driving data," *IEEE Trans. Intell. Vehicles*, vol. 2, no. 4, pp. 308–320, Dec. 2017.

[5] J. Gonçalves Filho, A. Patel, B. Lopes Alcantara Batista, and J. Celestino, "A systematic technical survey of DTN and VDTN routing protocols," *Comput. Standards Interfaces*, vol. 48, pp. 139–159, Nov. 2016.

[6] Y. Mao, C. Zhou, Y. Ling, and J. Lloret, "An optimized probabilistic delay tolerant network (DTN) routing protocol based on scheduling mechanism for Internet of things (IoT)," *Sensors*, vol. 19, no. 2, p. 243, Jan. 2019.

[7] A. Hamza-Cherif, K. Boussetta, G. Diaz, and F. Lahfa, "Performance evaluation and comparative study of main VDTN routing protocols under small-and large-scale scenarios," *Ad Hoc Netw.*, vol. 81, pp. 122–142, Dec. 2018.

[8] K. Bylykbashi, E. Spaho, L. Barolli, and F. Xhafa, "Routing in a many-to-one communication scenario in a realistic VDTN," *J. High Speed Netw.*, vol. 24, no. 2, pp. 107–118, Mar. 2018.

[9] G. Demin, L. Haifeng, L. Yunfei, and W. Guoxin, "Maximum data collection rate in rechargeable wireless sensor networks with multiple sinks," *China Commun.*, vol. 13, no. 2, pp. 95–108, Feb. 2016.

[10] F. Zhang, J. Thiyagalingam, T. Kirubarajan, and S. Xu, "Speed-adaptive multi-copy routing for vehicular delay tolerant networks," *Future Gener. Comput. Syst.*, vol. 94, pp. 392–407, May 2019.

[11] H. Lin and F. Zhang, "A scheme for stimulating message relaying cooperation," *Int. J. Distrib. Sensor Netw.*, vol. 16, no. 2, Feb. 2020, Art. no. 155014772090703.

[12] D. Tian, J. Zhou, Y. Wang, G. Zhang, and H. Xia, "An adaptive vehicular epidemic routing method based on attractor selection model," *Ad Hoc Netw.*, vol. 36, pp. 465–481, Jan. 2016.

[13] J. Cui, S. Cao, Y. Chang, L. Wu, D. Liu, and Y. Yang, "An adaptive spray and wait routing algorithm based on quality of node in delay tolerant network," *IEEE Access*, vol. 7, pp. 35274–35286, 2019.

[14] D. Gao, H. Lin, and X. Liu, "Routing protocol for k-anycast communication in rechargeable wireless sensor networks," *Comput. Standards Interfaces*, vol. 43, pp. 12–20, Jan. 2016.

[15] N. Kawabata, Y. Yamasaki, and H. Ohsaki, "On message delivery delay of epidemic DTN routing with broadcasting ACKs," in *Proc. IEEE 41st Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, vol. 1, Jul. 2017, pp. 701–704.

[16] H. Huang, S. Guo, P. Li, and T. Miyazaki, "Stochastic analysis on the deactivation-controlled epidemic routing in DTNs with multiple sinks," *Ad Hoc Sensor Wireless Netw.*, vol. 38, nos. 1–4, pp. 143–167, 2017.

[17] B. B. Bista and D. B. Rawat, "Enhancement of PRoPHET routing in delay tolerant networks from an energy prospective," in *Proc. IEEE Region 10 Conf. (TENCON)*, Nov. 2016, pp. 1579–1582.

[18] M. Hu, L. Luo, L. Kou, and H. Huang, "Message forwarding mechanism based on node behavior analysis in selfishness delay tolerant network," *Comput. Eng.*, vol. 11, p. 23, Apr. 2016.

[19] J. A. F. F. Dias, J. J. P. C. Rodrigues, N. Kumar, and C. X. Mavromoustakis, "A hybrid system to stimulate selfish nodes to cooperate in vehicular delay-tolerant networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Dec. 2015, pp. 5910–5915.

[20] D. K. Sharma, S. K. Dhurandher, I. Woungang, R. K. Srivastava, A. Mohananey, and J. J. P. C. Rodrigues, "A machine learning-based protocol for efficient routing in opportunistic networks," *IEEE Syst. J.*, vol. 12, no. 3, pp. 2207–2213, Sep. 2018.

[21] Y.-H. Xu, C.-C. Yang, M. Hua, and W. Zhou, "Deep deterministic policy gradient (DDPG)-based resource allocation scheme for NOMA vehicular communications," *IEEE Access*, vol. 8, pp. 18797–18807, 2020.

[22] B. Luo, D. Liu, T. Huang, and D. Wang, "Model-free optimal tracking control via critic-only Q-Learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 10, pp. 2134–2144, Oct. 2016.

[23] Y. Shen, N. Zhao, M. Xia, and X. Du, "A deep Q-Learning network for ship stowage planning problem," *Polish Maritime Res.*, vol. 24, no. s3, pp. 102–109, Nov. 2017.

[24] Y.-H. Xu, J.-W. Xie, Y.-G. Zhang, M. Hua, and W. Zhou, "Reinforcement learning (RL)-based energy efficient resource allocation for energy harvesting-powered wireless body area network," *Sensors*, vol. 20, no. 1, p. 44, Dec. 2019.

[25] T. Hester, "Deep q-learning from demonstrations," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 3223–3230.

[26] Z. Zhong, J. Yan, W. Wu, J. Shao, and C.-L. Liu, "Practical block-wise neural network architecture generation," 2017, *arXiv:1708.05552*. [Online]. Available: http://arxiv.org/abs/1708.05552

[27] Y. Li, "Deep reinforcement learning," 2018, *arXiv:1810.06339*. [Online]. Available: http://arxiv.org/abs/1810.06339

[28] F. Zhang, M. Shi, X. Yang, X. Tan, X. Qian, and I. Joe, "A reputation based intermediate message forwarding scheme," in *Proc. 5th Int. Conf. Electr., Electron., Comput. Sci.*, 2016, pp. 24–26.

[29] Van Hasselt, Hado, Arthur Guez, and David Silver, "Deep reinforcement learning with double q-learning," in *Proc. 13th AAAI Conf. Artif. Intell.*, 2016, pp. 2094–2100.

[30] Z. Zhong, Z. Yang, B. Deng, J. Yan, W. Wu, J. Shao, and C.-L. Liu, "BlockQNN: Efficient block-wise neural network architecture generation," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jan. 23, 2020, doi: 10.1109/TPAMI.2020.2969193.

[31] Z. Tong, H. Chen, X. Deng, K. Li, and K. Li, "A scheduling scheme in the cloud computing environment using deep Q-learning," *Inf. Sci.*, vol. 512, pp. 1170–1191, Feb. 2020.

[32] C. Wu, T. Yoshinaga, D. Bayar, and Y. Ji, "Learning for adaptive anycast in vehicular delay tolerant networks," *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 4, pp. 1379–1388, Apr. 2019.

[33] M. S. Hossen, M. T. Ahmed, and M. S. Rahim, "Effects of buffer size and mobility models on the optimization of number of message copies for multi-copy routing protocols in scalable delay-tolerant networks," in *Proc. Int. Conf. Innov. Sci., Eng. Technol. (ICISET)*, Oct. 2016, pp. 1–4.

[34] B. B. Bista, "Improving energy consumption of epidemic routing in delay tolerant networks," in *Proc. 10th Int. Conf. Innov. Mobile Internet Services Ubiquitous Comput. (IMIS)*, Jul. 2016, pp. 278–283.

[35] Z. Chen and C. Chen, "Self-stopping epidemic routing in cooperative wireless mobile sensor networks," in *Proc. IEEE 12th Int. Conf. Wireless Mobile Comput., Netw. Commun. (WiMob)*, Oct. 2016, pp. 1–7.

[36] S. Sati, A. Sohoub, A. Eltahar, K. A. B. Ahmad, K. Ahmad, and A. Bakeer, "Degree contact DC-epidemic routing protocol for opportunistic networks," in *Proc. Adv. Wireless Opt. Commun. (RTUWO)*, Nov. 2018, pp. 198–203.

[37] G. ur Rehman, A. Ghani, M. Zubair, S. Ahmad Khan Ghayyure, and S. Muhammad, "Honesty based democratic scheme to improve community cooperation for IoT based vehicular delay tolerant networks," 2020, *arXiv:2004.08698*. [Online]. Available: http://arxiv.org/abs/2004.08698

FUQUAN ZHANG received the M.S. degree in computer science from Shenyang Li Gong University in 2005 and the Ph.D. degree from Hanyang University, Seoul, South Korea. He is currently a Faculty Member of Nanjing Forestry University. His research interests include 3G/4G cellular systems and wireless mesh networks.

INWHEE JOE (Member, IEEE) received the B.S. and M.S. degrees in electronics engineering from Hanyang University, Seoul, South Korea, and the Ph.D. degree in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 1998. Since 2002, he has been a Faculty Member of the Division of Computer Science and Engineering, Hanyang University. His current research interests include wireless sensor networks, 3G/4G cellular systems, mobility management, multimedia networking, and performance evaluation.

HAIFENG LIN received the B.E. degree in computer science from Nanjing University, Nanjing, China, in 2003, the M.S. degree in computer engineering from the Memorial University of Newfoundland, St. John's, NL, Canada, in 2006, and the Ph.D. degree from Nanjing Forestry University, Nanjing. He is currently an Associate Professor with the Department of Computer Science, Nanjing Forestry University. His main research interests include wireless sensor networks, network technology, and the Internet of Things.

WANGUO JIAO received the B.S. degree in network engineering from Shijiazhuang Tiedao University, Shijiazhuang, China, in 2009, and the Ph.D. degree in telecommunication engineering from Xidian University, Xi'an, China, in September 2015. Since 2015, she has been with the College of Information Science and Technology, Nanjing Forestry University, where she is currently an Associate Professor. Her research interests include the performance analysis and protocol design of multi-hop wireless networks, cognitive radio networks, and wireless sensor networks.

YUZHEN DONG is currently pursuing the master's degree with Nanjing Forestry University, Nanjing. His main research interests include VDTN protocol and the Internet of Things.

YAN ZHANG received the B.E. degree in computer science from Nanjing Tech University, Nanjing, China, in 2005, and the M.S. and Ph.D. degrees from the Nanjing University of Aeronautics and Astronautics, Nanjing, in 2008 and 2015, respectively. He is currently a Lecturer with the Department of Computer Science, Nanjing Forestry University. His main research interests include process algebra, temporal logic, and formal verification.

● ● ●