RLID-V: Reinforcement Learning-Based Information Dissemination Policy Generation in VANETs

Yingjie Xia, *Member, IEEE*, Xuejiao Liu, Jing Ou, and Oubo Ma

***Abstract*— Ciphertext policy attribute-based encryption (CP-ABE) is popularly used to implement secure and accurate access control of disseminated information in vehicular ad hoc networks (VANETs). Nevertheless, how to improve the policy generation of CP-ABE for accurate information dissemination in the dynamic VANETs remains a challenge, as there are several access control policies rising from moving vehicles and road side units (RSUs) with different sensing boarder regarding to a specific event, such as moving vehicles and road side units (RSUs). To solve this problem, this paper proposes a reinforcement learning-based information dissemination policy generation scheme in VANETs, named RLID-V. The scheme firstly combines multiple attribute-based access control policies and resolves policy conflicts between vehicles and RSUs. Then, a manual feedback policy construction method is designed by applying decision tree to the collected feedback from all receivers. Finally, we employ reinforcement learning to dynamically update the confidence weights of different policy sources. The experiments are conducted in two classic VANETs scenarios, traffic guidance and accident warning, demonstrating that RLID-V achieves better performance in the accuracy and effectiveness of information dissemination compared with three existing schemes. Otherwise, RLID-V outperforms the compared schemes in robustness with 20% error feedback and takes a negligible cost of less than 1% of the overall delay overhead for policy generation.**

***Index Terms*— VANETs, Information Dissemination, Policy Combination, Reinforcement Learning.**

I. Introduction

INFORMATION dissemination plays an important role in vehicular ad hoc networks (VANETs), especially in road safety and traffic management applications [1], [2], [3], ciphertext policy attribute-based encryption (CP-ABE) has been widely studied for achieving secure and accurate information dissemination in VANETs [4], [5], [6]. Therefore, the accuracy for information dissemination is totally determined by the access control policies. Through the expressive and fine-grained access control policy over the information, it guarantees that only authorized users whose attributes match the access policy can access the encrypted information [7].

Most existing schemes assume that the access control policy could be designed by message senders (e.g., moving vehicles) [8], [9] by default. However, due to the restricted sensing capability, the access control policies enforced by the vehicles may be not accurate enough. Compared with the moving vehicles, road side units (RSUs) collect traffic information nearby to have a broader sensing range. Intuitively, how to combine the access control policies formulated by RSU and vehicles remains a challenging problem.

Due to the complex road conditions and communication environment, it is difficult to combine the access control policies to adapt to dynamic environment of VANETs. Firstly, fixed weights can not adapt different scenarios in VANETs. Then, RSU may keep a relatively higher impact on the combined policy for information with the larger region of interest (ROI). Also the combination scheme should have the ability to explore and adjust dynamically. Several artificial intelligence-based works in VANETs provide a reference [10], [11], [12], [13] that the process of dynamically adjusting confidence weights can be modeled as a markov decision process (MDP) and solved with reinforcement learning algorithms.

To address this problem, this paper proposes a reinforcement learning-based policy generation scheme called RLID-V for accurate information dissemination in VANETs. Specifically, the RSU generates original policy according to the message type and receives the encrypted message with its policy. Then, RSU utilizes these two policies to obtain the final combined policy. After the information dissemination is completed, RSU collects manual feedback from receivers and constructs a feedback policy. Finally, RSU uses the similarity between the combined policy and the feedback policy as a reward to guide the updating of the confidence weights. Compared with the existing studies, RLID-V autonomously optimizes policy combinations without requiring additional expert knowledge, making it adaptable to complex and dynamic scenarios in VANETs. Our contributions are given as follows.

(1) To achieve the policy combination in complex scenarios, we propose a context-based conflict resolution method which can resolve three types of conflicts for multiple policies from different entities in VANETs.

(2) To evaluate the effectiveness of the combined policy, we propose to construct a manual feedback policy in which the RSU collects feedback from receivers, uses a decision tree to

Manuscript received 25 August 2022; revised 8 July 2023; accepted 26 July 2023. Date of publication 16 August 2023; date of current version 29 November 2023. This work was supported in part by the Natural Science Foundation of Zhejiang Province under Grant LZ22F030004. The Associate Editor for this article was A. H. Sodhro. *(Corresponding author: Yingjie Xia.)*

Yingjie Xia is with the College of Computer Sciences, Zhejiang University, Hangzhou 310012, China, and also with Hangzhou Yuantiao Technology Company, Hangzhou 311112, China (e-mail: xiayingjie@zju.edu.cn).

Xuejiao Liu is with the School of Information Science and Technology, Hangzhou Normal University, Hangzhou 310030, China.

Jing Ou and Oubo Ma are with the College of Computer Sciences, Zhejiang University, Hangzhou 310012, China.

Digital Object Identifier 10.1109/TITS.2023.3300948

select vehicle attributes, and calculates the information entropy of the feedback.

(3) To iteratively improve accuracy of the policy combination, we propose a reinforcement learning-based confidence weight update method, in which the immediate reward is determined by the similarity between the combined policy and the manual feedback policy.

The rest of the paper is organized as follows. We discuss the related work in Section II. In Section III, we introduce the system model. Section IV describes proposed scheme in detail. The analysis of experiments is described in Section V. Finally, we conclude our work in Section VI.

II. Related Work

This section presents an overview of related work about information dissemination, policy combination for accurate information dissemination, and reinforcement learning schemes in VANETs.

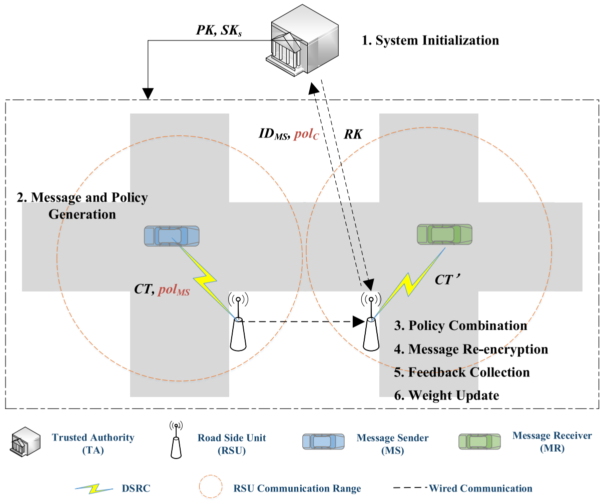


Fig. 1. The information dissemination scenarios in VAENTs and the six main steps included in RLID-V.

A. Information Dissemination in VANETs

In VANETs, information such as traffic accidents, road condition warnings, and location awareness services will be shared timely and securely. A vehicle constantly exchanges this available information with others and RSUs. Due to high mobility behavior and frequent temporary disconnection in vehicular communications, the reliable dissemination of information has become essential and challenging [14], [15]. More importantly, information dissemination should be secure and accurate to identify the target dissemination area, e.g., movement of vehicles in the directions [16]. In this paper, we focus on policy combination for accurate information dissemination in VANETs.

Due to the inherent openness in vehicular communication, many security threats are arising during information dissemination, including eavesdropping, tampering, and suppression [17]. Researchers focus on different encryption mechanisms to provide data confidentiality [18], [19]. Raya et al. utilized public key infrastructure to encrypt the message [20]. Federrath et al. utilized the Diffie-Hellman algorithm to generate a symmetric key and encrypt data between RSU and vehicle [21]. Lightweight symmetric cryptography was proposed by Zhu [22] to encrypt the communication data. Huang et al. first introduced ciphertext policy attribute-based encryption (CP-ABE) [23] in VANETs and proposed an attribute-based policy framework for fine-grained access control [24]. Compared with public key infrastructure methodologies [8], [9], [25], attribute-based encryption shows an attractive approach designed for ensuring one-to-many fine-grained access control of encrypted data.

B. Policy Combination for Access Control

Access control mechanisms are classical security issues, which are widely used to restrict unauthorized access to resources [26]. Various access control models have been proposed in the literature, such as role-based access control (RABC) [27]. However, the central architecture of RBAC is not suitable for today's mobile environment since roadside units are not trusted, and the vehicles are not restricted in the same environment [28].

Attribute-based access control [29] has gained attention in the distributed system with multiple administrative domains. Researches have addressed the effective method to improve attribute-based encryption to support the access control model, such as constructing flexible attribute-based data access control for cloud storage [30], incorporating user-specific privacy references to implement attribute-based security policies [31]. These researches provide fine-grained access control for the disseminated information to vehicles but ignore the conflicts between the policies designed by different entities.

C. Reinforcement Learning-Based Schemes in VANETs

Reinforcement learning has been used to improve the adaptability of different schemes to the rapid movement of vehicles and the dynamic changes of road environment in VANETs. Guo et al. [10] proposed a reinforcement learning model that allowed vehicles to adjust the trust evaluation strategy in different driving scenarios. Lu et al. [11] applied reinforcement learning to select the authentication modes and parameters in the physical authentication scheme to resist rogue edge attackers. Xiao et al. [32] designed a hotbooting policy hill climbing-based unmanned aerial vehicle (UAV) relay strategy with reinforcement learning to mitigate smart jamming in VANETs. Wu et al. [33] introduced a routing protocol for urban VANETs, which combined the advantages of geographic routing with the static road map and learned the road segment traffic information based on Q-learning algorithm.

III. Architecture

In this section, we outline the system architecture and implementation process of our scheme. There are four entities in the information dissemination scenario in VANETs: trusted authority (TA), road side unit (RSU), message sender (MS), and message receiver (MR). RLID-V realizes the

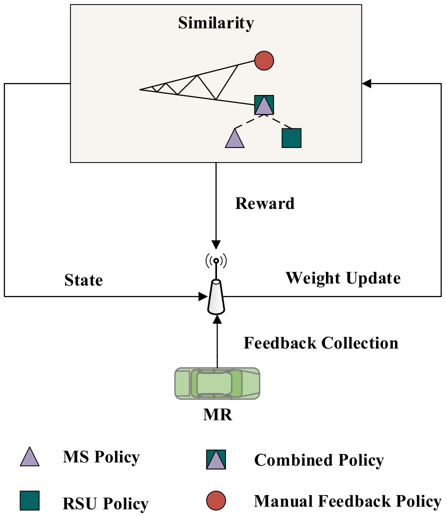


Fig. 2. RSU collects the feedback provided by the message receivers and then uses reinforcement learning to model and automatically update the confidence weights.

encrypted information dissemination based on the ciphertext-policy attribute-based proxy re-encryption (CP-ABPRE) [34], and its specific process is shown in Fig. 1. We will provide a concise description of these six steps.

1) System Initialization: TA executes the $Setup(1^{\lambda}, U)$ algorithm to generate the system public key PK and the master secret key MSK with the security parameter λ and system attributes U as input. Subsequently, TA distributes the public key PK to RSU and all vehicles. For each vehicle, TA employs the $KeyGen(PK, MSK, S)$ algorithm to generate the secret key SK_S and issues it via a secure channel, where S is the attribute set of the vehicle.

2) Message and Policy Generation: In the event of an accident or congestion, MS generates a message m and employs the $Enc(PK, pol_{MS}, m)$ algorithm to produce a ciphertext CT. Here, pol_{MS} represents the access control policy formulated by MS, structured as a disjunctive normal form (DNF) composed of the conjunctive normal form (CNF). The access control policy (pol) is defined as:

$$pol = \bigvee_{i=1}^{n_{CNF}} \left(\bigwedge AttrVal_i\right), \tag{1}$$

where n_{CNF} denotes the number of the CNF, $\bigvee$ represents the logical OR operation, $\bigwedge$ represents the logical AND operation, and $AttrVal$ denotes the attribute predictive value. The attribute predictive value $(AttrVal)$ is defined as $Na \otimes Vr$, where Na represents the attribute name, $\otimes$ denotes the relational operator, and $\otimes \in \{>, <, =, \geq, \leq\}$, while Vr represents the value range. The value range (Vr) can encompass both discrete and continuous values.

For instance, MS can define a policy of $(speed < 40) \bigvee (type = bus)$ for encrypting the message, only if the buses with the speed below 40 km/h can decrypt it. Subsequently, MS transmits the ciphertext (CT) and the policy (pol_{MS}) to the RSU.

3) Policy Combination: The RSU formulates its own access control policy, denoted as pol_{RSU}, based on its perception capabilities. Subsequently, RSU combines pol_{MS} and pol_{RSU} to create a composite policy, referred to pol_C. In cases where conflicts arise between pol_{MS} and pol_{RSU}, RSU employs a context-based conflict resolution method to resolve these conflicts and generate a new composite policy, denoted as pol_C. Following the policy construction, RSU transmits the MS identifier (ID_{MS}) and pol_C to the TA, requesting a re-encryption key. More details of the policy combination can be found in Section IV-A.

4) Message Re-Encryption: TA retrieves the secret key SK_{MS} associated with MS using the provided identifier ID_{MS}. Subsequently, TA executes the $ReKeyGen(PK, SK_{MS}, pol_C)$ algorithm to generate a re-encryption key RK, which is then transmitted to RSU. Upon receiving the re-encryption key, RSU employs the $ReEnc(PK, RK, CT)$ algorithm to produce a re-encrypted ciphertext denoted as CT'. The re-encrypted ciphertext CT' is then disseminated within the communication range of RSU.

Upon receiving the ciphertext message, MRs execute the $Dec(PK, SK_{MR}, CT')$ algorithm. If the attribute sets of the MRs satisfy the access policy pol_C specified for CT', the algorithm successfully decrypts the ciphertext and outputs the plaintext message m.

5) Feedback Collection: After receiving the message, all MRs engage in providing manual feedback, which is subsequently uploaded to RSU. RSU gathers the manual feedback from the MRs and leverages this data to construct a manual feedback policy. The construction of the manual feedback policy involves considering information entropy and decision tree techniques, which enable RSU to effectively analyze and derive insights from the collected feedback. More details of the feedback collection can be found in Section IV-B.

6) Weight Update: As illustrated in Fig. 2, RSU employs a reinforcement learning approach to enhance the accuracy of the combined policy. This iterative feedback-driven update mechanism contributes to refining and improving the accuracy of the combined policy over time. More details of the weight update can be found in Section IV-C.

pol_{MS}, ID_{MS}, pol_C, and RK are transmitted as ciphertext, utilizing a separate public-private key system independent of CP-ABPRE. The sender encrypts the message using the receiver's public key, and the receiver subsequently decrypts the ciphertext using their private key. This additional encryption mechanism ensures secure transmission and confidentiality of the mentioned components during the communication process.

IV. Implementation of RLID-V

This section gives the implementation details of three parts of the overall scheme, policy combination by context-based conflict resolution, feedback collection and reinforcement learning-based weight update.

A. Context-Based Conflict Resolution

The most prominent problem in policy combination is policy conflict. We assume that there are two existing policies pol_A and pol_B and define the attributes in pol_A and pol_B as two

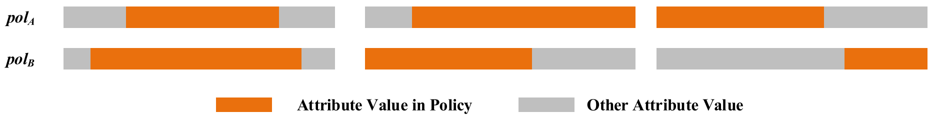


Fig. 3. Three types of conflicts between different policies: cover (left), overlap (middle), non-overlap (right).

sets Na^A and Na^B. If an attribute Na_i exists in both Na^A and Na^B, and the value Vr_i differs between the policies pol_A and pol_B, then policy conflict arises. As shown in Fig. 3, there are three types of conflicts.

- *Cover.* Both pol_A and pol_B contain this attribute, and the value range of pol_B covers the value range of pol_A.
- *Overlap.* Both pol_A and pol_B contain this attribute, and their value ranges partially overlap.
- *Non-overlap.* Both pol_A and pol_B contain this attribute, but their value ranges do not overlap.

We define the confidence weight pair of pol_A and pol_B as (ρ_A, ρ_B), where $\rho_A + \rho_B = 1$. For an attribute, the value range of pol_A is $[a_l, a_r]$, and the value range of pol_B is $[b_l, b_r]$. Three conflict resolution methods are given below (see Fig. 4).

1) Cover Resolution: We multiply the minimum/maximum attribute values set in the two policies by their confidence weights respectively, the value range of the attribute in the combined policy is $[\rho_A a_l + \rho_B b_l, \rho_A a_r + \rho_B b_r]$.

2) Overlap Resolution: Similar to the cover resolution, the value range of the attribute in this case is also $[\rho_A a_l + \rho_B b_l, \rho_A a_r + \rho_B b_r]$.

3) Non-Overlap Resolution: The final value range for this case consists of two parts. The first part is taken from policy A with a smaller value, we keep the maximum value unchanged as a_r and set the minimum value as $\rho_A a_l + (1 - \rho_A)a_r$. The second part is taken from policy B with a larger value, we keep the minimum value unchanged as b_l and set the maximum value as $(1 - \rho_B)b_l + \rho_B b_r$. Finally, the value range of the attribute in the combined policy is $[\rho_A a_l + (1 - \rho_A)a_r, a_r] \cup [b_l, (1 - \rho_B)b_l + \rho_B b_r]$.

Algorithm 1 gives the implementation of conflict resolution.

B. Feedback Collection

MRs will respond to RSU whether the information is valid. Considering that there exist some MRs that may send wrong feedback intentionally or unintentionally. RSU constructs the manual feedback policy by feedback data. It will be utilized during the dynamic update phase of the confidence weights.

1) Remove Abnormal Feedback: We remove abnormal feedback which is different from the feedback with similar vehicles. The similarity of two vehicles can be judged by their attribute vectors. The attribute vector is defined as

$$\vec{v} = (Ea_1, Ea_2, Da_1, Da_2), \tag{2}$$

where Ea_1 and Ea_2 are the static attributes, such as vehicle type, license plate and manufacturer; Da_1 and Da_2 are the dynamic attributes, such as speed, driving direction, geographic location and road conditions. For example, RSU generates a vector $\vec{v_i} = (0, 0, 1, 10)$ for a vehicle v_i according to Table. I, where vehicle v_i is a *bus* manufactured by manufacturer M_A and is currently traveling on $Road_B$ at 30 km/h. This table can be predefined by TA during the initialization phase and adjusted according to requirements.

Algorithm 1 Policy Combination

```
Input: Two original policies $pol_A$ and $pol_B$, the confidence weight pair $(\rho_A, \rho_B)$.
$Na^C \leftarrow Na^A \cap Na^B$
$|Na^C| \leftarrow n^C_{CNF}$
For $i = 1, 2, \ldots, n^C_{CNF}$ do:
    If $Na_i \in Na^A$ and $Na_i \in Na^B$ then:
        If there is a cover or overlap conflict between $Vr_i^A$ and $Vr_i^B$ then:
            $Vr_i^C \leftarrow [\rho_A a_l + \rho_B b_l, \rho_A a_r + \rho_B b_r]$
        If there is a cover non-overlap conflict between $Vr_i^A$ and $Vr_i^B$ then:
            $Vr_i^C \leftarrow [\rho_A a_l + (1-\rho_A)a_r, a_r] \cup [b_l, (1-\rho_B)b_l + \rho_B b_r]$
    If $Na_i \in Na^A$ and $Na_i \notin Na^B$ then:
        $Vr_i^C \leftarrow [a_l, a_r]$
    If $Na_i \notin Na^A$ and $Na_i \in Na^B$ then:
        $Vr_i^C \leftarrow [b_l, b_r]$
$pol_C \leftarrow \bigvee_{i=1}^{n^C_{CNF}} (\bigwedge AttrVal_i)$
Output: The combined policy $pol_C$.
```

TABLE I

EXAMPLE OF A VECTOR GENERATION TABLE

Attribute	Value
Vehicle type	$bus(0); truck(1); taxi(2)$
Manufacturer	$M_A(0); M_B(1); M_C(2)$
Location	$Road_A(0); Road_B(1); Road_C(2)$
Speed(m/s)	[0,15]

In addition, all attributes need to be normalized as follows.

$$Attr = \frac{Attr - Attr_{min}}{Attr_{max} - Attr_{min}}, \tag{3}$$

where $Attr \in \{Ea_1, Ea_2, Da_1, Da_2\}$ denotes vehicle attributes; $Attr_{max}$ and $Attr_{min}$ are the maximum and minimum values of the value range of $Attr$.

The similarity of vehicle v_i and vehicle v_j is defined as

$$sim(v_i, v_j) = \frac{\vec{v_i} \cdot \vec{v_j}}{|\vec{v_i}| \times |\vec{v_j}|}. \tag{4}$$

Vehicles with a similarity higher than T_{sim} are clustered into a group, where $T_{sim} \in [0, 1]$ is a predefined similarity threshold.

2) Construct Manual Feedback Policy: RSU utilizes the feedback to construct a manual feedback policy. Firstly, RSU divides the vehicles into two groups: the message is considered valid G_v and the message is considered invalid G_i. RSU utilizes a decision tree algorithm to select vehicle attributes

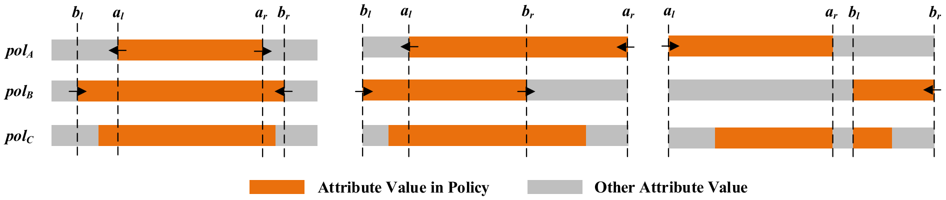


Fig. 4. Three types of conflict resolution: cover resolution (left), overlap resolution (middle), non-overlap resolution (right).

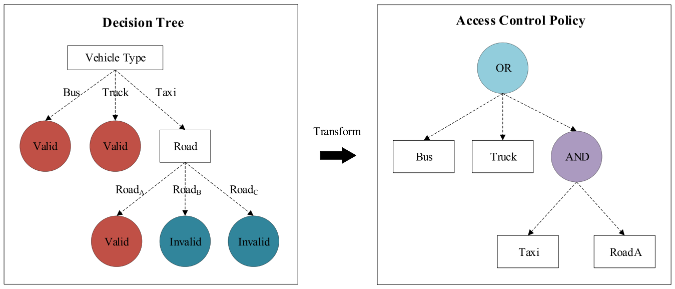


Fig. 5. Transformation from a decision tree to a policy.

and calculates the information entropy of the feedback.

$$H = -(plog_2 p + (1-p)log_2(1-p)), \tag{5}$$

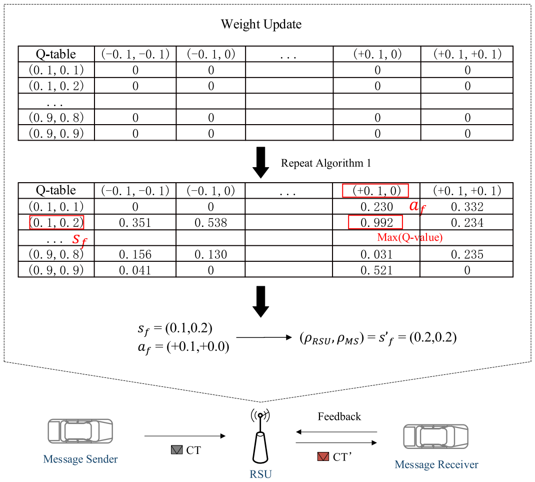
where p is the proportion of vehicles in G_v. RSU selects the attribute with the largest information gain ratio for division. By selecting attributes recursively, RSU obtains the final decision tree. We define T_{deep} as the maximum depth. If the depth of the decision tree is smaller than T_{deep} and all data samples can be classified into a unique leaf node, the decision tree is constructed. Otherwise, RSU stops the recursion and set the label of each leaf node as the majority data label.

Upon constructing the final decision tree, RSU transforms it into a formatted attribute-based policy. Firstly, RSU extracts the leaf nodes labeled as successfully decrypted and their root-leaf paths. The non-leaf nodes in each path are connected with logic AND to construct a sub-policy. Finally, RSU constructs the manual feedback policy by combing these sub-policies with logic OR. As shown in Fig. 5, the decision tree shows that buses, trucks, or taxis driving on the $Road_A$ regarding the message are valid. The root-leaf paths are $\{bus\}$, $\{truck\}$, and $\{taxi, Road_A\}$ and the final policy should be $bus \bigvee truck \bigvee (taxi \bigwedge Road_A)$.

C. Reinforcement Learning-Based Weight Update

RSU updates the confidence weight pair (ρ_{MS}, ρ_{RSU}) based on reinforcement learning to improve the similarity between the combined policy and the manual feedback policy. The updated (ρ_{MS}, ρ_{RSU}) can be implemented offline and will be used for accurate information dissemination next time.

1) Policy Similarity: We assume that policy $pol_A = \bigvee_{i=1}^{m}(pol_{A_i})$ has m sub-policies and $pol_B = \bigvee_{j=1}^{n}(pol_{B_j})$ has n sub-policies. pol_{A_i} and pol_{B_j} are sub policies which utilize logic AND to connect attribute and its value. To calculate the similarity between these two policies, we need to construct a mapping from pol_A to pol_B. In the mapping, each sub policy in pol_A is mapped to one sub policy in pol_B, vice versa. If the number of two policies are not the same, we supplement it with an empty policy. Then, we can find the optimal mapping relationship to maximize the sum of similarity of the paired sub-policies $max(\sum_{max(m,n)} sim(pol_i, pol_j))$, where $sim(pol_i, pol_j)$ is the similarity of pol_{A_i} and pol_{B_j}, which is defined as


Weight Update

Q-table	(−0. 1, −0. 1)	(−0. 1, 0)	. . .	(+0. 1, 0)	(+0. 1, +0. 1)
(0. 1, 0. 1)	0	0		0	0
(0. 1, 0. 2)	0	0		0	0
. . .					
(0. 9, 0. 8)	0	0		0	0
(0. 9, 0. 9)	0	0		0	0

Repeat Algorithm 1

Q-table	(−0. 1, −0. 1)	(−0. 1, 0)	. . .	(+0. 1, 0)	(+0. 1, +0. 1)
(0. 1, 0. 1)	0	0		0. 230	0. 332
(0. 1, 0. 2)	0. 351	0. 538		0. 992	0. 234
. . .					
(0. 9, 0. 8)	0. 156	0. 130		0. 031	0. 235
(0. 9, 0. 9)	0. 041	0		0. 521	0

$s_f = (0.1,0.2)$
$a_f = (+0.1,+0.0)$ → $(\rho_{RSU}, \rho_{MS}) = s'_f = (0.2,0.2)$

Fig. 6. An example of RSU using reinforcement learning to update confidence weights.

$$sim(pol_{A_i}, pol_{B_j}) = \frac{\sum_{k=1}^{num_{ca}} sim(Attr_k^A, Attr_k^B)}{\left|pol_{A_i} \cup pol_{B_j}\right|}, \tag{6}$$

where $\left|pol_{A_i} \cup pol_{B_j}\right|$ is the number of attributes exists in the two sub policies; num_{ca} is the number of common attributes in pol_{A_i} and pol_{B_j}; $sim(Attr_k^A, Attr_k^B)$ is the similarity of the same attribute $Attr_k$ in the two sub-policies, which is calculated as

$$sim(Attr_k^A, Attr_k^B) = \frac{\left|Attr_k^A \cap Attr_k^B\right|}{\left|Attr_k^A \cup Attr_k^B\right|}. \tag{7}$$

It is a classic bipartite graph optimal matching problem and can be solved by Kuhn-Munkras algorithm [35]. Finally, the similarity between pol_A and pol_B is

$$sim(pol_A, pol_B) = \frac{max(\sum_{max(m,n)} sim(pol_i, pol_j))}{max(m,n)}. \tag{8}$$

2) Weight Update Method: RSU utilizes reinforcement learning to update the confidence weight pair (ρ_{MS}, ρ_{RSU}) every time it generates a manual feedback policy. This process can be modeled as an markov decision process (MDP) formalized as a tuple

$$\mathcal{G} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma), \tag{9}$$

Algorithm 2 Reinforcement Learning-Based Weight Update

```
**Input:** The confidence weight $(\rho_{MS}, \rho_{RSU})$.
Generate the manual feedback policy $pol_O$
$s \leftarrow (\rho_{MS}, \rho_{RSU})$
Initialize $Q(s, a), num_{set}$
Repeat (for each episode):
  Initialize state $s$, $num_{repeat} = 0$
  Repeat (for each step of episode):
    Select action $a$
    Calculate $s' = s + a$
    if $s' = terminal$:
      break
    Calculate $\Delta sim = sim_{s'} - sim_s$
    if $\Delta sim > 0$:
      $r = 1$
    else:
      $r = 0$
    $Q(s, a) \leftarrow Q(s, a) + \alpha[r + \gamma max_{a'} Q(s', a') - Q(s, a)]$
    $sim_s \leftarrow sim_{s'}$
    $s \leftarrow s'$
    $num_{repeat} \leftarrow num_{repeat} + 1$
  Until $num_{repeat} > num_{set}$
$(\rho_{MS}, \rho_{RSU}) \leftarrow s$
**Output:** The updated confidence weight $(\rho_{MS}, \rho_{RSU})$.
```

where $\mathcal{S} = \{(s_1, s_2)|s_1 \in \{0.1, \ldots, 0.9\}, s_2 \in \{0.1, \ldots, 0.9\}\}$ denotes the state space. $\mathcal{A} = \{(a_1, a_2)|a_1 \in \{0.0, \pm 0.1\}, a_2 \in \{0.0, \pm 0.1\}\}$ denotes the action space. $\mathcal{P} : \mathcal{S} \times \mathcal{A} \rightarrow \Delta(\mathcal{S})$ represents the probability that taking an action $a \in \mathcal{A}$ in state $s \in \mathcal{S}$. $\mathcal{R} : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ denotes the reward function. $\gamma \in [0, 1]$ is the discount factor which assigns a higher importance to immediate rewards. Further, we utilize the confidence weight pair as the row index and the actions as the column index to get a Q-table, which is stored in the RSU and initialized with all zeros.

RSU executes the weight update algorithm (Algorithm 2) for a fixed number of episodes each time, and an episode ends when $s' = terminal$ or the number of explorations num_{repeat} exceeds the preset num_{set}. At the beginning of each episode, s is randomly selected from $\mathcal{S}$, and $num_{repeat} = 0$ is reset. RSU utilizes $\epsilon - greedy$ as the exploration strategy, which uses $\epsilon \in [0, 1]$ as parameter of exploration to decide which action to perform using $Q(s, a)$. RSU selects an action a from $\mathcal{A}$ with the highest Q-value in the current state with probability $1 - \epsilon$, and a random action otherwise. The next state s' can be obtained through the current state s and the selected action a. For example, if $s = (0.5, 0.6)$, $a = (+0.1, -0.1)$, then $s' = (0.6, 0.5)$.

If $s' \neq terminal$, RSU calculates the similarity difference $\Delta sim = sim_{s'} - sim_s$, where sim_s is the similarity between the combined policy pol_C in state s and the manual feedback policy; $sim_{s'}$ is the similarity between the combined policy $pol_{C'}$ in state s' and the manual feedback policy. The calculation method can refer to Eq(8). If $\Delta sim > 0$, it means that the similarity between the two policies is improved. In this situation we set the reward $r = 1$, otherwise $r = 0$. Then, RSU updates the Q-value in each exploration as follows.

TABLE II

VEHICLE PARAMETERS

Vehicle Type	Length(m)	Average speed(m/s)	Proportion(%)
car	4	10	70
bus	10	8	10
truck	12	6	20

$$Q(s, a) = Q(s, a) + \alpha[r + \gamma max_{a'} Q(s', a') - Q(s, a)], \tag{10}$$

where $\alpha \in [0, 1]$ is the learning rate. Finally, RSU updates sim_s as $sim_{s'}$, s as s', and num_{repeat} as $num_{repeat} + 1$.

After completing the fixed episode of updates, RSU will look for the largest Q-value in the final Q-table, and get the corresponding row and column indices (state s_f and action a_f). As shown in Fig. 6, RSU utilizes s_f and a_f to get s'_f as the final result to update (ρ_{MS}, ρ_{RSU}).

V. PERFORMANCE ANALYSIS

This section describes the evaluation setup, scenarios, metrics, and comparison scenarios. Then, we evaluate RLID-V in terms of accuracy, effectiveness, robustness, and cost.

A. Setup

Similar to [10], [36], and [37], we utilize SUMO [38] to simulate realistic vehicle movement. Then we utilize libfenc cryptographic library [39] to realize the ciphertext-policy attribute-based proxy re-encryption (CP-ABPRE), which adopts a 224-bit elliptic curve bilinear keystore from Stanford University [40]. We run our experiments using Java and python in Windows operating system with Intel(R) Core(TM) i5-4200U @1.60GHz CPU and 4G RAM.

For the reinforcement learning-based weight update, we set the discount factor $\gamma = 0.9$, the total number of the episode as 500, and the total number of explorations num_{set} as 100 in the simulations. We perform an extensive grid-search to find the best parameter combination (α, ϵ) for the reinforcement learning-based weight update method [41]. The range of α and ϵ is [0.05, 0.75], and the step size is 0.05. We test all parameter combinations within the range and output the result as the average similarity between the final combined policy and the manual feedback policy. The parameter combination with the largest average similarity will be regarded as the feedback parameter combination. As shown in Fig. 7, the similarity result and the learning rate α are negatively correlated. According to the results, we choose $\alpha = 0.05$ and $\epsilon = 0.05$ as the parameters of subsequent experiments.

B. Scenarios and Metrics

Traffic guidance and accident warning are common and important applications in VANETs. Therefore, we choose these two scenarios to evaluate our scheme.

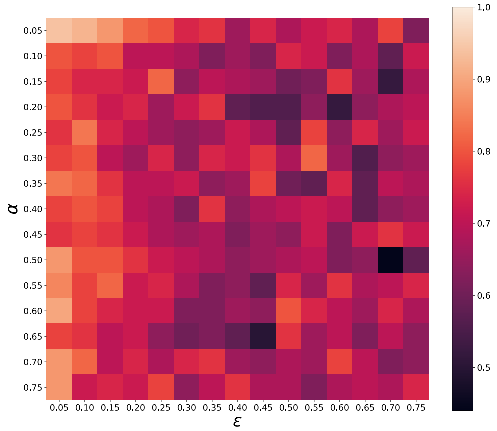
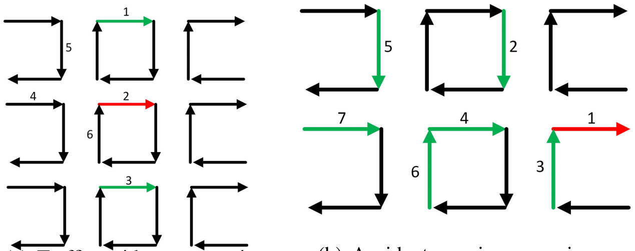


Fig. 7. Heatmap of the combination of parameters. The horizontal coordinate corresponds to the exploration parameter and the vertical coordinate corresponds to the learning rate of reinforcement learning.



(a) Traffic guidance scenario

(b) Accident warning scenario

Fig. 8. Two classic VANETs scenarios.

1) Traffic Guidance Scenario: Traffic guidance information needs to send the guidance route to the vehicles within a specific range, so that they can change the driving path, thereby reducing the potential risk of congestion. A small number of vehicles guided to the target road cannot effectively alleviate congestion. On the contrary, the excessive number of vehicles guided to the target road may cause new congestion.

Fig. 8(a) is an example in the simulation where there is a congestion on $road_2$. Traffic information is disseminated to vehicles driving on $road_4$, $road_5$ and $road_6$, which assists the vehicles in selecting appropriate routes $road_1$ and $road_3$. The vehicle parameters are given in Table. II. The length of each road is 500m, and the speed limit is 15m/s. The induction obedience rate is set to be 70%, which means that the vehicle may not change its driving path after receiving the message. In this scenario, the attributes in the policy include $\{type, length, speed, location\}$.

Generally, the degree of congestion can directly measure the effect of traffic guidance, so we define the following three metrics.

$$avp = \frac{\sum_{i \in num_v} speed_i}{num_v}, \tag{11}$$

where avp is the average speed of all vehicles; num_v is the number of vehicles; $speed_i$ is the speed of vehicle v_i.

$$ocd = \frac{\sum_{i \in num_r} \sum_{j \in num_r, j \neq i} |occ_i - occ_j|}{C^2_{num_r}}, \tag{12}$$

where $ocd \in [0, 1]$ indicates whether the traffic distribution between the roads is uniform; num_r is the number of roads; $occ_i \in [0, 1]$ is the ratio of the number of vehicles on $road_i$ to the upper limit; $C^2_{num_r}$ is the number of combinations to take 2 elements out of num_r different elements.

$$score_{tg} = \omega_1 \cdot avp + \omega_2 \cdot (1 - ocd), \tag{13}$$

where $score_{tg} \in [0, 1]$ is the comprehensive metric, which measures the accuracy of the policy; ω_1 and ω_2 are the weights of average speed and occupancy difference, and $\omega_1 + \omega_2 = 1$. In the simulation, we set them to be 0.5.

2) Accident Warning Scenario: When there is an accident on the road, it is necessary to broadcast the warning information to surrounding roads as soon as possible. If the formulated disseminating range is too small, the potentially affected vehicles will not be able to plan a more reasonable driving path early because they cannot obtain the information of the road ahead in time. On the contrary, if the formulated disseminating range is too large, vehicles that are irrelevant and meet the policy will receive and decrypt the message. This will cause additional burdens and even mislead these vehicles to take a detour.

Fig. 8(b) is an example in the simulation. There is an accident in $road_1$, and the congestion spreads upstream roads(e.g. $road_2$ to $road_7$) quickly. We simulate the accident by blocking lanes. Each road can normally pass 5,000 vehicles per hour, but this number drops to 2,500 when blocked. The driving direction of the vehicle is simplified as $\{up, down, left, right\}$. In this scenario, the attributes in the policy include $\{speed, location, direction\}$.

Accidents are difficult to simulate directly, so we assume that potentially affected vehicles will not collide if they successfully receive and decrypt the accident warning information, otherwise there is a higher risk of an accident. Whether the vehicle will be affected can be judged according to the simulated driving path when there is no accident warning. Based on the above analysis, we define three metrics.

$$pre = \frac{num_{ra}}{num_{re}}, \tag{14}$$

where pre is the precision of policy; num_{ra} is the number of vehicles that received the message and were affected by congestion. num_{re} is the number of vehicles that received the message.

$$cov = \frac{num_{ra}}{num_{af}}, \tag{15}$$

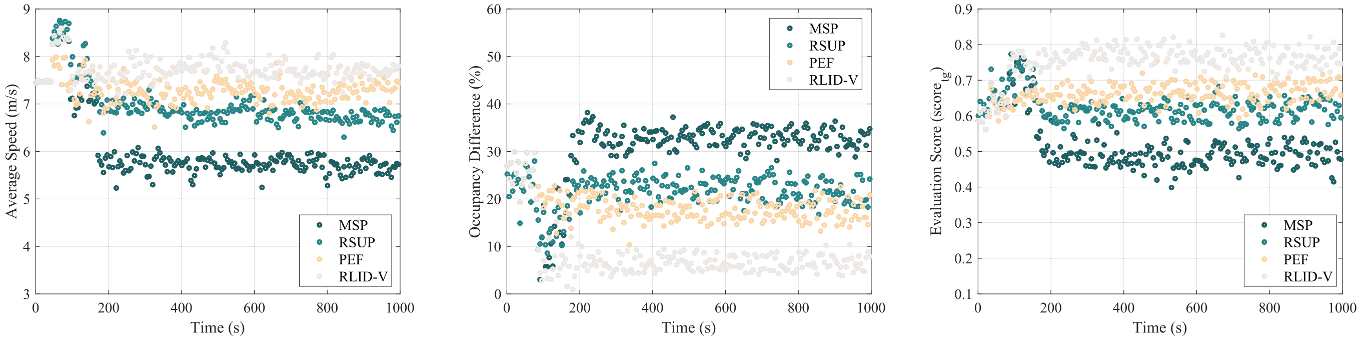
where cov is the coverage of policy; num_{af} is the number of vehicles affected by congestion.

$$score_{aw} = 2 \cdot \frac{pre \cdot cov}{pre + cov}, \tag{16}$$

where $score_{aw} \in [0, 1]$ is the comprehensive metric, which measures the accuracy of the policy.

C. Comparison With the Existing Schemes

We compare RLID-V with three other schemes: (1) MSP: we extract the part of the MSP that implements information dissemination based on CP-ABE, where the access policy is



(a) Average Speed

(b) Occupancy Difference

(c) Evaluation Score

Fig. 9. The accuracy of information dissemination in the traffic guidance scenario.

set by the message sender [9]. (2) RSUP: the access policy in MSP is designed individually by the RSU, which follows the standard implementation of CP-ABE in VANETs [25]. (3) PEF [42]: the access policy is jointly formulated by the message sender and the RSU.

D. Simulation Results

We evaluate the performance of RLID-V in four dimensions: accuracy, effectiveness, robustness, and cost. To avoid fluctuation, we get the average results from 100 times simulations.

1) Accuracy Analysis: As shown in Fig. 9(a), in the traffic guidance scenario, the performance of our scheme tends to be stable when the simulation reaches about 100s, and the average speed of all vehicles fluctuates up and down at 7.7m/s. Compared with other schemes, the average speed of all vehicles in our scheme is the highest, which means that vehicles get accurate traffic guidance. As shown in Fig. 9(a), our scheme can stabilize the occupancy difference below 10%, which is much lower than the other three schemes. This shows that the vehicles are more evenly distributed on different roads under the traffic guidance, which can effectively alleviate the congestion. As shown in Fig. 9(c), the average evaluation score $score_{tg}$ is about 0.51, 0.62, 0.66, and 0.75 under MSP, RSUP, PEF, and our scheme. The results show that our scheme can improve the accuracy of the combined policy in traffic guidance scenarios to select more suitable vehicles for guidance.

As shown in Fig. 10, in the accident warning scenario, the results of the three metrics of our scheme are 0.93, 0.95 and 0.94, respectively. The results show that our scheme can set a more accurate disseminating range in the combined policy to help vehicles plan more reasonable driving paths.

2) Effectiveness Analysis: To prove the effectiveness of the combined policy obtained when policies pol_{MS} and pol_{RSU} have large deviations, this paper randomly changes the attribute values of these two policies for simulation. In the traffic guidance scenario, the attribute values we can change are the road and vehicle type. We set the number of changed roads as num_{cr} and set the number of changed vehicle types as num_{ct}. As show in Fig. 8(a), we only consider the road segments next to the congested road, and the number of vehicle types is three so that $num_{cr} \leq 3$ and $num_{ct} \leq 3$. Therefore, we assume that $num_{cr} = num_{ct} = 3$ in pol_{MS}.

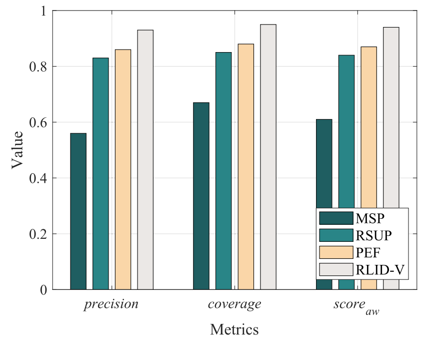
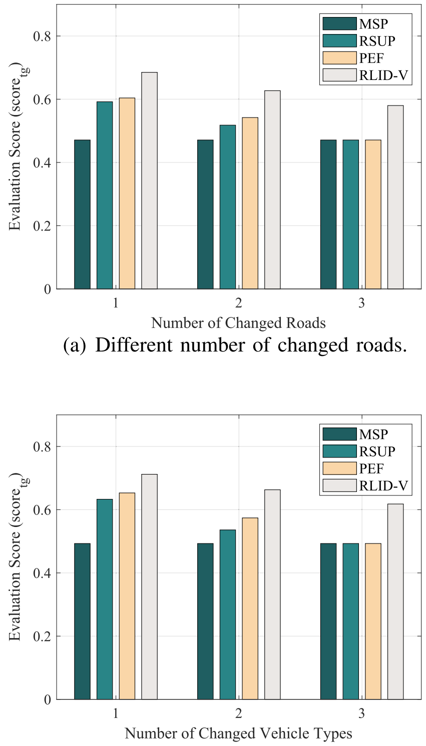


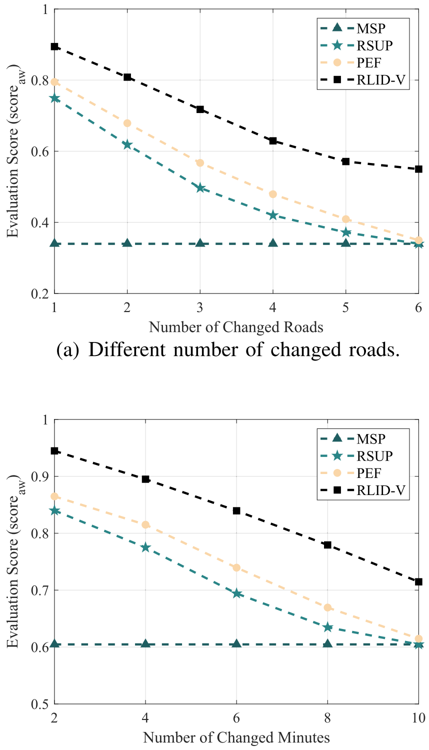
Fig. 10. The accuracy of information dissemination in the accident warning scenario.



(a) Different number of changed roads.

(b) Different number of changed vehicle types.

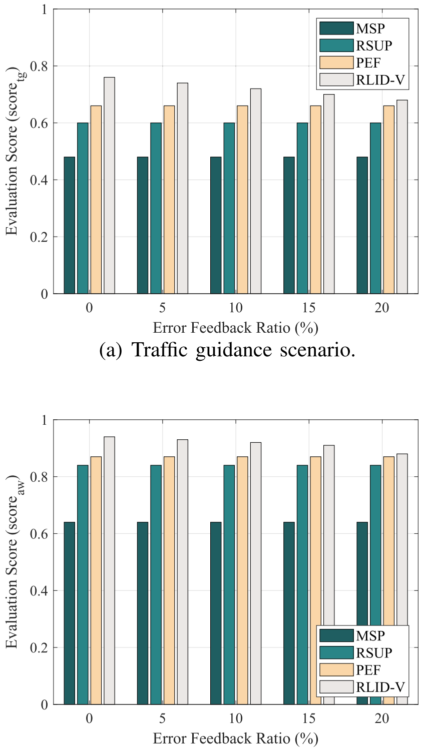
Fig. 11. The effectiveness of information dissemination in the traffic guidance scenario.



(a) Different number of changed roads.

(b) Different number of changed minutes.

Fig. 12. The effectiveness of information dissemination in the accident warning scenario.



(a) Traffic guidance scenario.

(b) Accident warning scenario.

Fig. 13. Robustness comparison under different proportions of abnormal feedback.

As shown in Fig. 8, as num_{cr} and num_{ct} increase, evaluation scores of all policies decrease because of the larger deviations between designed policies and manual feedback policy. When $num_{cr} = 3$ or $num_{ct} = 3$, the evaluation score of PEF is almost the same as MSP and RSUP because it can not gain more useful information from the disseminated information with bad policies. However, the evaluation score of RLID-V is the largest in any case because it can maximize the similarity between the combined policy and the manual feedback policy.

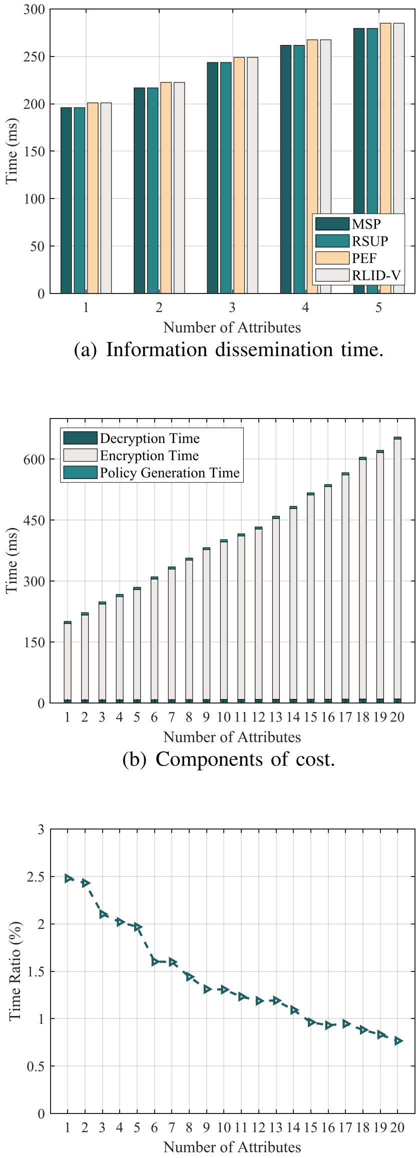
In the accident warning scenario, we set the number of roads as num_{br} and the number of minutes as num_{bm}. As show in Fig. 8(b), we only consider roads from $road2$ to $road7$ in this scenario, so that $num_{br} \leq 6$. In addition, we set $num_{bm} \in \{2, 4, 6, 8, 10\}$.

As shown in Fig. 12, the score of all policies decrease as the number of changed roads or changed minutes grows because of larger deviations between designed policies and manual feedback policy. When the deviation between pol_{MS} and pol_{RSU} is large, PEF is not able to increase the evaluation score anymore. Due to the feedback-based adjustment ability, RLID-V has better performance than other schemes.

3) Robustness Analysis: RSU structures the manual feedback policy based on feedback from MRs. Assume that some MRs respond to wrong feedback intentionally. As shown in Fig. 13, the evaluation score decreases when the proportion of wrong feedback data grows. When the proportion of wrong feedback increases to 20%, the scores of RLID-V decrease by 0.06 and 0.08 in the two scenarios, which is still better than the other three schemes. This is because in our scheme, RSU detects and removes some abnormal data at the beginning. Therefore, the combined policy is error-tolerant with feedback from MRs, which enhances its practicality.

4) Cost Analysis: In VANETs, delay cost is another important metric for measuring information dissemination schemes. The total cost for information dissemination including all policy generation time, encryption time, and decryption time. It should be noted that RSU uses reinforcement learning to update the confidence weights offline, and this part of the time overhead does not bring additional delay to information dissemination.

Fig. 14(a) shows that RLID-V does not cause a significant increase in transmission time compared with other schemes. As shown in Fig. 14(b), the total information dissemination time of RLID-V increases linearly with the growing number of attributes. The policy generation time ratio among information dissemination time is shown in Fig. 14(c). Policy generation time ratio denotes the ratio of the policy generation time to the overall scheme execution time. With the increasing number of attributes, the encryption time increases rapidly due to the complex paring computation of the attributes, with a 9.8% average increasing rate, but our policy generation time only increases relatively lower, with a 0.01% average increasing rate. Therefore, the policy generation time ratio decreases as the number of attributes increases. When the number of attributes is 20, the ratio drops to a minimum of 0.77%. This shows that the policy combination can



(a) Information dissemination time.

(b) Components of cost.

(c) Policy generation time ratio of RLID-V.

Fig. 14. Cost analysis for policy generation.

improve the accuracy while incurring only a small additional cost.

VI. Conclusion

This paper proposes a reinforcement learning-based policy combination scheme called RLID-V for accurate information dissemination in VANETs. Specifically, we propose that RSU assist vehicles in formulating policies to improve the accuracy of information dissemination. A conflict resolution method is utilized to deal with three kinds of policy conflicts to ensure the normal generation of combined policies. We also propose that the RSU collects feedback from the receivers to generate a feedback policy and adaptively adjusts the confidence weights with reinforcement learning. Experiments demonstrate that RLID-V enhances the accuracy and effectiveness of information dissemination in two classic VANETs scenarios compared with three schemes. In addition, RLID-V exhibits robustness to abnormal feedback and introduces a negligible cost of just 0.77% of the overall delay overhead for policy generation with 20 attributes. In current design of RLID-V, the emphasis lies on the specific utilization of conjunction, disjunction, and some-of structures within the policy tree. In our future work, we intend to explore more general policy structure that offer broader applicability and flexibility.

References

[1] R. Shrestha, R. Bajracharya, A. P. Shrestha, and S. Y. Nam, "A new type of blockchain for secure message exchange in VANET," *Digit. Commun. Netw.*, vol. 6, no. 2, pp. 177–186, May 2020.

[2] S. S. Moni and D. Manivannan, "A scalable and distributed architecture for secure and privacy-preserving authentication and message dissemination in VANETs," *Internet Things*, vol. 13, pp. 1–21, Mar. 2021.

[3] N. A. Zardari, R. Ngah, O. Hayat, and A. H. Sodhro, "Adaptive mobility-aware and reliable routing protocols for healthcare vehicular network," *Math. Biosci. Eng.*, vol. 19, no. 7, pp. 7156–7177, 2022.

[4] X. Liu, W. Chen, and Y. Xia, "Security-aware information dissemination with fine-grained access control in cooperative multi-RSU of VANETs," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 3, pp. 2170–2179, Mar. 2022.

[5] W. Luo and W. Ma, "Efficient and secure access control scheme in the standard model for vehicular cloud computing," *IEEE Access*, vol. 6, pp. 40420–40428, 2018.

[6] A. H. Sodhro et al., "Towards 5G-enabled self adaptive green and reliable communication in intelligent transportation system," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 8, pp. 5223–5231, Aug. 2021.

[7] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *Proc. Int. Workshop Public Key Cryptogr.* Cham, Switzerland: Springer, 2011, pp. 53–70.

[8] X. Liu, Y. Xia, W. Chen, Y. Xiang, M. M. Hassan, and A. Alelaiwi, "SEMD: Secure and efficient message dissemination with policy enforcement in VANET," *J. Comput. Syst. Sci.*, vol. 82, no. 8, pp. 1316–1328, Dec. 2016.

[9] J. Ma, T. Li, J. Cui, Z. Ying, and J. Cheng, "Attribute-based secure announcement sharing among vehicles using blockchain," *IEEE Internet Things J.*, vol. 8, no. 13, pp. 10873–10883, Jul. 2021.

[10] J. Guo et al., "TROVE: A context-awareness trust model for VANETs using reinforcement learning," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6647–6662, Jul. 2020.

[11] X. Lu, L. Xiao, T. Xu, Y. Zhao, Y. Tang, and W. Zhuang, "Reinforcement learning based PHY authentication for VANETs," *IEEE Trans. Veh. Technol.*, vol. 69, no. 3, pp. 3068–3079, Mar. 2020.

[12] A. H. Sodhro, G. H. Sodhro, M. Guizani, S. Pirbhulal, and A. Boukerche, "AI-enabled reliable channel modeling architecture for fog computing vehicular networks," *IEEE Wireless Commun.*, vol. 27, no. 2, pp. 14–21, Apr. 2020.

[13] A. Lakhan, M. A. Dootio, T. M. Groenli, A. H. Sodhro, and M. S. Khokhar, "Multi-layer latency aware workload assignment of e-transport IoT applications in mobile sensors cloudlet cloud networks," *Electron.*, vol. 10, no. 14, pp. 1719–1743, 2021.

[14] M. U. Ghazi, M. A. K. Khattak, B. Shabir, A. W. Malik, and M. S. Ramzan, "Emergency message dissemination in vehicular networks: A review," *IEEE Access*, vol. 8, pp. 38606–38621, 2020.

[15] O. Ma, X. Liu, and Y. Xia, "ABM-V: An adaptive backoff mechanism for mitigating broadcast storm in VANETs," *IEEE Trans. Veh. Technol.*, vol. 72, no. 7, pp. 8886–8897, Jul. 2023.

[16] D. Tian et al., "A distributed position-based protocol for emergency messages broadcasting in vehicular ad hoc networks," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 1218–1227, Apr. 2018.

[17] A. M. Vegni and V. Loscrí, "A survey on vehicular social networks," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2397–2419, 4th Quart., 2015.

[18] A. H. Sodhro, S. Pirbhulal, M. Muzammal, and L. Zongwei, "Towards blockchain-enabled security technique for industrial Internet of Things based decentralized applications," *J. Grid Comput.*, vol. 18, no. 4, pp. 615–628, Dec. 2020.

[19] M. Ali, L. T. Jung, A. H. Sodhro, A. A. Laghari, S. B. Belhaouari, and Z. Gillani, "A confidentiality-based data classification-as-a-service (C2aaS) for cloud security," *Alexandria Eng. J.*, vol. 64, pp. 749–760, Feb. 2023.

[20] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *J. Comput. Secur.*, vol. 15, no. 1, pp. 39–68, 2007.

[21] K. Plößl and H. Federrath, "A privacy aware and efficient security infrastructure for vehicular ad hoc networks," *Comput. Standards Interface*, vol. 30, no. 6, pp. 390–397, Aug. 2008.

[22] X. Zhu, Y. Lu, X. Zhu, and S. Qiu, "Lightweight and scalable secure communication in VANET," *Int. J. Electron.*, vol. 102, no. 5, pp. 765–780, May 2015.

[23] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. 13th ACM Conf. Comput. Commun. Secur.*, Oct. 2006, pp. 89–98.

[24] D. Huang and M. Verma, "ASPE: Attribute-based secure policy enforcement in vehicular ad hoc networks," *Ad Hoc Netw.*, vol. 7, no. 8, pp. 1526–1535, Nov. 2009.

[25] S.-J. Horng, C.-C. Lu, and W. Zhou, "An identity-based and revocable data-sharing scheme in VANETs," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 15933–15946, Dec. 2020.

[26] R. Tourani, S. Misra, T. Mick, and G. Panwar, "Security, privacy, and access control in information-centric networking: A survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 566–600, 1st Quart., 2018.

[27] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role-based access control models," *Computer*, vol. 29, no. 2, pp. 38–47, 1996.

[28] F. Li, Y. Rahulamathavan, M. Conti, and M. Rajarajan, "Robust access control framework for mobile cloud computing network," *Comput. Commun.*, vol. 68, pp. 61–72, Sep. 2015.

[29] R. Bobba et al., "Attribute-based messaging: Access control and confidentiality," *ACM Trans. Inf. Syst. Secur.*, vol. 13, no. 4, pp. 1–35, Dec. 2010.

[30] Y. Zhu, D. Huang, C.-J. Hu, and X. Wang, "From RBAC to ABAC: Constructing flexible data access control for cloud storage services," *IEEE Trans. Services Comput.*, vol. 8, no. 4, pp. 601–616, Jul. 2015.

[31] M. Gupta, F. M. Awaysheh, J. Benson, M. Alazab, F. Patwa, and R. Sandhu, "An attribute-based access control for cloud enabled industrial smart vehicles," *IEEE Trans. Ind. Informat.*, vol. 17, no. 6, pp. 4288–4297, Jun. 2021.

[32] L. Xiao, X. Lu, D. Xu, Y. Tang, L. Wang, and W. Zhuang, "UAV relay in VANETs against smart jamming with reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4087–4097, May 2018.

[33] J. Wu, M. Fang, H. Li, and X. Li, "RSU-assisted traffic-aware routing based on reinforcement learning for urban vanets," *IEEE Access*, vol. 8, pp. 5733–5748, 2020.

[34] H. Li and L. Pang, "Efficient and adaptively secure attribute-based proxy reencryption scheme," *Int. J. Distrib. Sensor Netw.*, vol. 12, no. 5, pp. 1–12, 2016.

[35] X. Dai, X. Wang, and N. Liu, "Optimal scheduling of data-intensive applications in cloud-based video distribution services," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 1, pp. 73–83, Jan. 2017.

[36] H. E. Sayed, S. Zeadally, and D. Puthal, "Design and evaluation of a novel hierarchical trust assessment approach for vehicular networks," *Veh. Commun.*, vol. 24, pp. 1–11, Aug. 2020.

[37] X. Liu, O. Ma, W. Chen, Y. Xia, and Y. Zhou, "HDRS: A hybrid reputation system with dynamic update interval for detecting malicious vehicles in VANETs," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 12766–12777, Aug. 2022.

[38] *Sumo Homepage*. Accessed: 2021. [Online]. Available: http://sumo.sourceforge.net/

[39] R. M. Green, A. Akinyele, and M. Rushanan. (2004). *Libfenc: The Functional Encryption Library*. Accessed: 2021. [Online]. Available: http://code.google.com/p/libfenc/

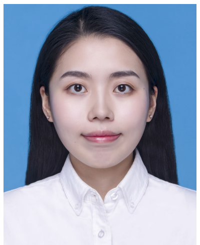
[40] *Stanford Pairings-Based Crypto Library*. Accessed: 2021. [Online]. Available: http://crypto.stanford.edu/pbc/

[41] A. D. Tijsma, M. M. Drugan, and M. A. Wiering, "Comparing exploration strategies for Q-learning in random stochastic Mazes," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Dec. 2016, pp. 1–8.

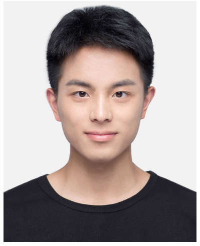
[42] Y. Xia, X. Liu, J. Ou, and W. Chen, "A policy enforcement framework for secure data dissemination in vehicular ad hoc network," *IEEE Trans. Veh. Technol.*, vol. 70, no. 12, pp. 13304–13314, Dec. 2021.

Yingjie Xia (Member, IEEE) received the Ph.D. degree from the College of Computer Science, Zhejiang University, in 2009. He was affiliated as a Research Scientist with the National Center for Supercomputing Applications (NCSA), University of Illinois at Urbana-Champaign (UIUC), USA. He is currently an Associate Professor with Zhejiang University and the CEO of Hangzhou Yuantiao Technology Company. His research interests include intelligent transportation and information security.

Xuejiao Liu received the Ph.D. degree in computer science from Huazhong Normal University, Wuhan, China. She is currently an Associate Professor with the School of Information Science and Technology, Hangzhou Normal University, Hangzhou, China. Her research interests include network security and data security.



Jing Ou received the M.S. degree in cyberspace security from Zhejiang University, Hangzhou, China. Her research interests include VANETs and information security.



Oubo Ma received the M.S. degree from Hangzhou Normal University, Hangzhou, China. He is currently pursuing the Ph.D. degree with Zhejiang University. His research interests include network security and reinforcement learning security.