Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 6942120, 18 pages
https://doi.org/10.1155/2022/6942120

Research Article

Joint Trust Management and Sharing Provisioning in IoV-Based Urban Road Network

Tao Jing, Yue Liu, Xiaoxuan Wang, and Qinghe Gao

School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China

Correspondence should be addressed to Xiaoxuan Wang; xiaoxuanw@bjtu.edu.cn

Received 17 December 2021; Accepted 18 February 2022; Published 14 March 2022

Academic Editor: Alessandro Bazzi

Internet of Vehicles (IoV) is a novel technology to enhance the safety, intelligence, and efficiency of traffic systems, where vehicles can exchange critical information with other vehicles, roadside units, pedestrians, and cloud platforms. However, the dynamic network topology, high speed, and exposed communication links inevitably pose security threats to IoV. It is pivotal to establish a trust management and trust-sharing mechanism between vehicles to guarantee the safety of IoV. This paper proposes a distributed trust management scheme to discriminate malicious vehicles utilizing the machine learning technology Random Forest (RF). With the help of the sliding time window technology, the trust degree of vehicles can be comprehensively evaluated through the CART trees according to the current and historical records. To further improve the security of communication processes, we also introduce a lightweight cryptography mechanism. In addition, a trust-sharing mechanism based on path prediction algorithm is proposed to guarantee the consistency of trust information in the network. Finally, extensive simulations are conducted to demonstrate the feasibility and efficiency of the proposed scheme.

1. Introduction

Under the facilitation of 5G/B5G, bulks of smart-apparatuses are connected to the Internet to execute massive information interactions, symbolizing the official arrival of Internet of Things (IoT) [1]. Internet of Vehicle (IoV) is a variation of IoT. The ultimate goal of IoV is to enhance the safety, intelligence, and efficiency of traffic systems, with vehicles exchanging critical information with other traffic entities. Recently, IoV is on the verge of widespread deployment with the emergence of various advancements in radio access and core network technologies [2, 3]. Equipped with intelligent devices, such as wireless sensors and On-Board Units (OBU), vehicles have powerful communication, storage, and computing capabilities. Besides, IoV is also capable of implementing the Intelligent Transportation System (ITS), and the integration of dynamic information service, which can reduce the number of traffic accidents and alleviate traffic congestions [4, 5].

Meanwhile, the highly dynamic network topology, unconversant relationships between vehicles, and exposed communication links inevitably pose security menaces to IoV. On one hand, there are deviations between the information obtained by vehicles and the natural environment because of the failures of sensors or other smart devices. On the other hand, malicious vehicles can acquire illegal benefits by injecting false information into the network. They can directly forge and broadcast fake messages, disguise as legitimate entities, and even tamper with the transmitter's practical information [6], which causes threats to the authenticity and reliability of the information. Based on the above analyses, it is particularly urgent to establish an efficient trust management mechanism for IoV. A proper trust management mechanism can discriminate malicious nodes, resist malicious attacks, and ensure the stability of communication processes, thereby improving driving conditions and ultimately improving the safety of IoV.

The concept of "trust management" was first proposed by M. Blaze in 1996 [7]. The author emphasized that trust management is an integral part of network service security. Trust management mechanisms can be considered from identity verification, attack detection and mitigation, confidentiality, privacy, trust and reputation, and other dimensions. The kernel of trust management is to formulate a

suitable trust evaluation mechanism according to precise regulations. Malicious nodes can be discriminated by calculating trust value, and then, other nodes in the network select trusted nodes for interactions.

According to the framework of trust management mechanisms, it can be divided into two categories: centralized and decentralized management. For centralized trust management, a trusted entity in the network is required to execute the trust management mechanism, and all information is reposited in a central server. When vehicles need to exchange information with other vehicles, they must communicate with the central server. Centralized trust management mechanism has good stability but poor scalability and cannot adapt to the highly dynamic network topology of IoV. Besides, it also faces a single point of failure problem. Conspicuous, the decentralized trust management is more applicable to IoV. At present, the researches on decentralized trust management mechanism mainly adopt the following underlying technologies: cryptographic, recommendation-based, fuzzy logic-based, game theory-based, and machine learning-based approach.

Cryptography is the first line of defense for communication systems. Choi et al. [8] first associate symmetric certification by using short-lived pseudonyms in VANETs. Vasudev et al. [9] propose a lightweight trust authentication and management scheme using Cryptographic Hash Functions, but it lacks the judgment on fake messages. In IoV, it is essential to ensure the trustworthiness of vehicles, but the authenticity of messages also cannot be fooled. Ahmad et al. [10] propose MARINE to detect and revoke dishonest vehicles, incorporating entity and data trust. Wang et al. [11] propose a distributed HDMA scheme for 5G-enabled VANETs using a group signature-based algorithm for mutual authentication between V2V communications.

The trust management method based on neighborhood vehicle recommendation is realized through indirect communication between vehicle nodes. Hu et al. [12] propose a scheme called "REPLACE," which is a trust-based platoon service recommendation scheme to help the user vehicles avoid choosing badly behaved platoon head vehicles. Ahmed et al. [13] combined direct and indirect trust to identify any potential malicious nodes in the current network by calculating local trust and analyzing suggestions from other neighbors. Li et al. [14] propose a reputation-based global trust establishment scheme (RGTE) that safely shares the trust information in VANET by applying statistical laws. In addition to the above two methods, Soleymani et al. [15] propose a fuzzy trust model based on experience and plausibility to secure the vehicular network. Guleng et al. [16] propose a scheme that uses a fuzzy logic-based trust calculation approach to evaluate the direct trust of trustee nodes. Halabi and Zulkernine [17] present a vehicular coalition formation approach that incorporates a hedonic cooperative game model, which aims at preventing malicious or faulty vehicles from joining collaborative benign vehicular communities. With the emergence of machine learning, scholars have made researches on the application of this in network security [18]. Jiang et al. [19] propose a new trust evaluation and update mechanism for underwater wireless sensor networks based on the C4.5 decision tree algorithm (TECU).

However, vehicle travels at high speed following the intended driving route and only establishing a suitable trust management mechanism that is not sufficient to ensure the safety of IoV. How to certify the consistency of "trust" is another problem worth paying attention to. At present, most researchers use the central system controller to share the trust value of the vehicles. However, the location of the central controller is generally stationary, and the potency of trust sharing drops markedly as the distance between the vehicle and the controller increases, which imposes restrictions on the scalability of IoV.

Considering the property of decentralization, immutability, transparency, and fault tolerance of blockchain, many researchers use blockchain technology to realize trust management and sharing mechanism. Singh et al. [20] propose a blockchain-based decentralized trust management scheme using smart contracts. Specifically, they introduce blockchain sharding to reduce the load on the main blockchain and increase the transaction throughput. Yang et al. [21] propose a traffic event validation and trust verification mechanism based on blockchain's decentralized nature and first proposes the "proof-of-event" consensus algorithm to ensure the correctness of stored information. However, it must be noted that the mining cost of running consensus mechanisms is expensive and requires enormous computing and storage resources, limiting blockchain applications.

Our article proposes a Random Forest-based trust management mechanism named MTRF for IoV to determine vehicles' identities and ensure vehicle network security. To avoid the overfitting problem for decision learning technology, we combine the ensemble learning method Random Forest (RF), which allows the model to limit overfitting without increasing the error due to bias. Besides, we also propose a trust-sharing mechanism based on a path prediction algorithm to forecast the following orientations of vehicles. The trust value of the vehicle can be shared point-to-point between RSUs to conquer the negative impact of the central controller. The main contributions of this paper are summarized as follows:

(i) To reduce the excessive network resources' consumption and the increased difficulty of vehicle management caused by dynamic vehicle topologies, we propose a dynamic clustering process according to their current locations, driving directions, and other parameters. We also adopt RF technology to realize cooperative multivehicle trust management in a temporary cluster to achieve malicious vehicle identification

(ii) Considering the conceptual nature of "trust," a single correct behavior is not enough to prove the vehicle's identity. Therefore, we introduce a sliding time window algorithm to store the vehicles' decision results at different time slots and comprehensively evaluate the degree of vehicle trust. In addition, we set a penalty factor to prevent sudden attacks from malicious nodes with higher accumulated trustworthiness

(iii) To secure communication links between Cluster-Member-vehicles (CMVs), Cluster-Head-Vehicles (CHVs), and RSUs, we introduce a lightweight

cryptography scheme based on Elliptic Curve Cryptography (ECC), Cryptographic Hash Function, and, XOR operations

(iv) To accurately share the trust information of vehicles during cluster-switching, we propose a trust-sharing mechanism by utilizing a DQN-based path prediction algorithm. Therefore, the trust information of the corresponding vehicle can be shared between RSUs to conquer the negative impact of the central controller and improve system scalability

The rest of this paper is organized as follows. The system model are presented in Section 2. Section 3 introduces the proposed trust management mechanism. The lightweight cryptography algorithm is presented in Section 4. Section 5 presents the trust sharing mechanism. Then, the performances of our proposed mechanisms are evaluated in Section 6. Finally, the conclusion is evaluated in Section 7

2. System Model

In this paper, we mainly consider the research on trust management and trust-sharing mechanism of IoV under the urban road network scenario. Figure 1 illustrates a typical urban road network architecture composed of numerous intersections, where vehicles are randomly deployed on the roads with known origins and destinations. Roadside units (RSUs) are deployed at the intersections along the roads. Three are two kinds of communication modes in the system: vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I), both of which can be undermined by attackers and reveal important information.

2.1. The Threat and Adversary Model. A variety of emerging communication technologies provide a stable connection between vehicles but also put forward higher requirements on the network model, communication protocol, quality of service, and security of communication system. Due to the strange relationship between vehicles, the authenticity and reliability of the message are questionable.

Figure 2 illustrates three major threats to IoV, in which *vehicle 1* observes the accident message and transmits to *vehicle 2* and *vehicle 3* [22]. Figure 2(a) is from the perspective of legitimate vehicles. It is assumed that *vehicles 1*, *2*, and *3* are legitimate vehicles, and the communication links are not attacked. However, intelligent devices such as sensors of *vehicle 3* are faulty. At this point, *vehicle 2* successfully receives the accurate information about the real event sent by *vehicle 1*, while *vehicle 3* receives an error message $mess\ age_{A'}$ because of sensor malfunction. Even if the identity of the vehicle is legal, it also unintentionally spreads false information into the network.

Figures 2(b) and 2(c) are for malicious vehicles. In 2(b), the malicious *vehicle 1* tampers with the observed accident information and acts as an information source to transmit false messages to other vehicles for deception. In 2(c), malicious *vehicle 2* communicates with *vehicle 1* as a legitimate entity to obtain accident information, tampers with and forwards the information to *vehicle 3*, and finally commits fraud by destroying the communication link between *vehicle 1* and *3*. They inject false information into the network to disrupt the transportation system and seek illegal profits. In our trust management mechanism, we mainly consider methods to resist the above two attack modes to resist malicious vehicle attacks.

2.2. Trust Management Process. To improve the framework's flexibility and implement the RF, we divide the vehicles into many clusters that mainly execute decision processes based on the communication processes between vehicles. Vehicles have two types: Cluster-Head-Vehicles (CHVs) and Cluster-Member-Vehicles (CMVs). CMVs establish communication links with other same-clustered vehicles and collect trust evidence to evaluate the identity of the node transmitting messages. The CHV selected for each cluster manages other same-clustered vehicles and communicates with RSUs. We set buses as CHVs to ensure high reliability, computing, and storage capacity [23]. It must be noted that clusters are temporary and updated overtime because of the high mobility of vehicles and the dynamic topology changes of communication networks [24].

Figure 3 visually depicts the configuration of a unitary intersection. The cluster regulated by CHV1 can illustrate the trust management process. If vehicle 1 observes the accident on the road, it will immediately generate the message and broadcast it in the cluster to inform other vehicles. CMVs can judge the trust identities of others in the cluster by utilizing RF algorithm based on the collected trust evidence. CMVs may have different decision results for the exact observed vehicle. CHV collects the decision results from the cluster and transmits them to RSU, enabling RSU to comprehensively consider the different decision results and decide the credibilities of vehicles by updating the vehicle's trust value based on the final integration result.

However, a high trust value of vehicle at the current time is not necessarily indicative of the reliable identification of the vehicle. Trust is a dynamic accumulated value that allows vehicles to use both current and historical records as benchmarks for the trust value. RSU adopts sliding time window technology to compute the trust value of the vehicle both based on the reputation value from the current and former time slots. RSU has the right to remove the vehicle whose trust value is below the specified threshold from the current cluster and notify other CHVs of the vehicle's identity.

By utilizing Random Forest, the identity of vehicle is jointly determined by the other vehicles in the cluster, which partly avoids the problem of decision failure caused by communication interruption between CMVs. However, CMVs, CHVs, and RSUs exchange highly-aggregated information, severely affecting the accuracy of vehicle identity judgment. To strengthen the mechanism's ability against attacks, we introduce a lightweight cryptography mechanism based on Elliptic Curve Cryptography (ECC), Cryptographic Hash Function, and XOR operations to protect the above communication processes from being destroyed by malicious vehicles.

2.3. Trust-Sharing Process. The driving routes of CHVs are comparatively fixed. The diversity of driving orientations

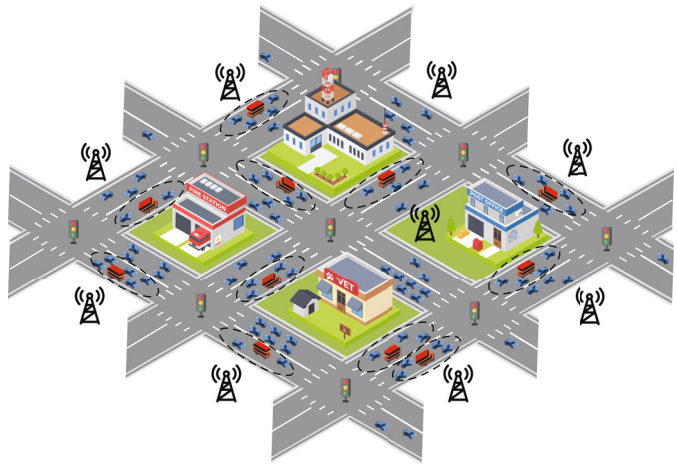
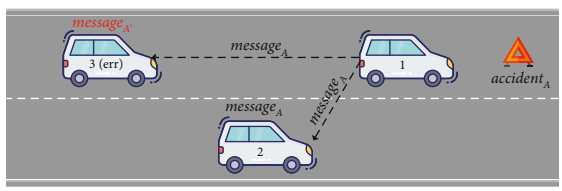
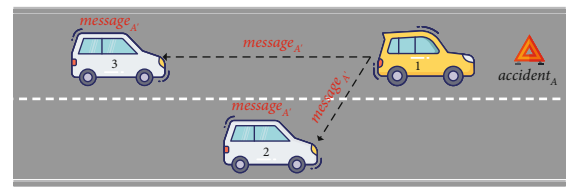


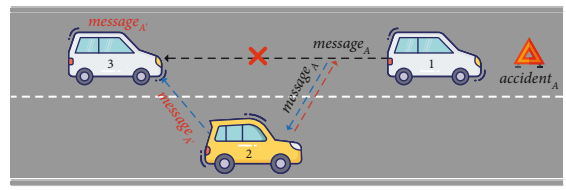
Figure 1: The system architecture of MTRF.



(a) Type 1



(b) Type 2



(c) Type 3

Figure 2: The threat and adversary model.

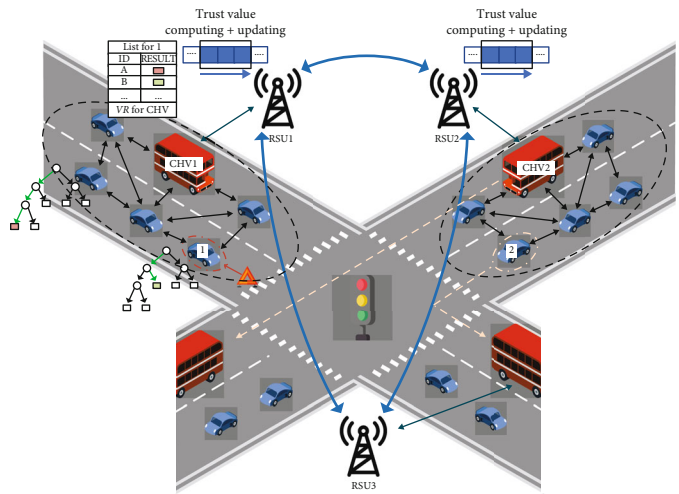


Figure 3: The Framework of a single cluster.

between CHV and CMV causes the vehicle to break away from the current cluster and find an appropriate one called cluster switching. Meanwhile, the trust information of vehicles needs to be synchronized to the corresponding CHV to facilitate the implementation of trust management for newcomers. To overcome the weakness of the traditional algorithm, we propose a novel trust-sharing mechanism based on the vehicle path prediction algorithm utilizing deep reinforcement learning in this paper.

Our algorithm takes the intersection as a unit. Vehicles execute the path predicting algorithm based on the traffic conditions to forecast the following driving orientations by Deep Q-network whenever they reach intersections. When the prediction is complete, CHV receives the prediction results sent by CMVs and compares the information with its direction. If there is a discrepancy, the CHV will establish a communication link with its RSU, and then RSU finds the applicable RSU in the same direction as the prediction. The vehicle trust information is transmitted between RSUs to facilitate synchronization to the corresponding CHVs. Vehicles can predict the next driving direction so that the trust value of the vehicle can be shared point-to-point between RSUs to conquer the negative impact of the central controller, which brings the benefits as follows:

(1) The path prediction algorithm is executed at the vehicle layer to improve network scalability and adapted to IoV

(2) The vehicle trust information is only transferred between CHVs and RSUs, without additional communication with the central controller, which reduces signaling overhead

(3) Vehicles need not to maintain communication links with ancient CHVs to reduce communication resource consumption. The target CHV receives vehicle trust information before the vehicle reaches the cluster and actively establishes communication when the vehicle enters its communication range.

3. Trust Management Process of MTRF

Figure 4 shows the primary process of the proposed trust management mechanism, which consists of five parts: dynamic clustering, trust evidence collection and preprocessing, trust evaluation, trust value calculation and update, and communication process encryption. In this section, we elaborate on each of the above four former parts.

3.1. Dynamic Clustering Process. The high-density vehicles are randomly deployed at intersections with different speeds and paths, which significantly increases the difficulty of vehicle management. An apposite dynamic clustering process is indispensable to reduce excessive signaling overhead and enhance the stability and scalability of the system.

The Euclidean distance between vehicle i and j is defined as d_{ij}. This parameter is collectively determined by the current location of the vehicle and the destination location. $X_i(t)$ and $\tilde{X}_i(t+\Delta t)$, respectively, represent the current and the estimated position of vehicle i. $X_i(t)$, and $\tilde{X}_i(t+\Delta t)$ and d_{ij} are expressed as follows:

$$\begin{aligned} X_i(t) &= (x_i, y_i), \\ \tilde{X}_i(t+\Delta t) &= \left(x_i + v_{i,x}\Delta t, y_i + y_{i,x}\Delta t\right), \\ d_{ij} &= \left\{ \left[\left(x_j + v_{j,x}\Delta t\right) - \left(x_i + v_{i,x}\Delta t\right)\right]^2 \right. \\ &\quad \left. + \left[\left(y_j + v_{j,y}\Delta t\right) - \left(y_i + v_{i,y}\Delta t\right)\right]^2 \right\}^{\frac{1}{2}}. \end{aligned} \tag{1}$$

To maintain the relative stability of a cluster, we also consider the driving directions of vehicles in our study. The driving direction of vehicle i is defined as d_i. Only the vehicles moving in the same directions can be grouped into a same cluster. Binary judgment variable α_{ij} is defined to describe this constraint.

$$\alpha_{ij} = \begin{cases} 1, & d_i = d_j \\ 0, & d_i \neq d_j \end{cases}, \tag{2}$$

where $\alpha_{ij} = 1$ represents vehicle i and j that have the same driving directions; otherwise, $\alpha_{ij} = 0$. After defining d_{ij} and α_{ij}, we can execute vehicle clustering operations. The bus w with R_w communication range is designated as the CHV of the $w-th$ cluster. CMVs belonging to cluster w can be defined as follows:

$$\beta_{wi} = \begin{cases} 1, & \alpha_{wi} d_{wi} \in [0, R_w) \\ 0, & \alpha_{wi} d_{wi} \notin [0, R_w) \end{cases}, \tag{3}$$

where β_{wi} represents whether vehicle i belongs to cluster w, and the CMV i will become the member of $m-th$ cluster only if it satisfies the requirement of driving condition and communication condition simultaneously.

Since the irregular distribution of CHVs, a CMV may be located at the overlapping area of two clusters. In this case, we stipulate that vehicle chooses the cluster where vehicle is closest to its corresponding CHV. We assume that vehicle j locates at the overlapping area of cluster w and $w+1$, and then, it chooses the cluster by

$$V_j \in \min\left(\frac{\alpha_{wj} d_{wj}}{R_w}, \frac{\alpha_{(w+1)j} d_{(w+1)j}}{R_{(w+1)}}\right). \tag{4}$$

As stated previously, clusters are impermanent and changing over time. If an CMV changes its path or takes an overtaking action, it will potentially exceed the current CHV's communication range and depart from the current cluster. In this case, to ensure the consistency of the vehicle trust value, the vehicle trust information needs to be synchronized with the corresponding CHV. This paper introduces a DQN-based path prediction algorithm for vehicles to solve the above issue, which will be described in Section 5.

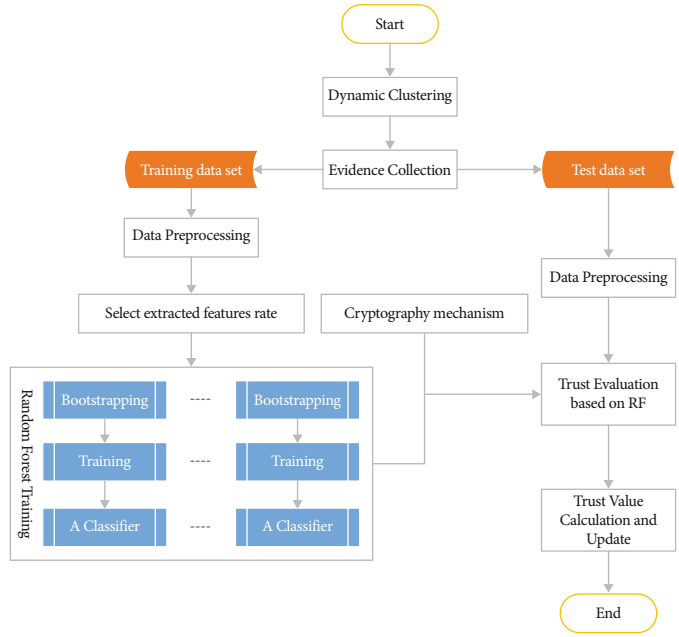


Figure 4: The overview of MTRF.

3.2. Trust Evidence Collection. Trust evidence is the basis of decision tree learning used to train the tree's structure and test the tree's accuracy, which is crucial to the performance of MTRF. We collect evidence from three aspects: vehicle-based, data-based, and link-based to consider the credibility of IoV comprehensively. However, the raw data collected by sensors contain missing values, outliers, and obsolete or redundant fields. To ensure the accuracy of the RF-based trust management mechanism, we must preprocess the trust evidence before training. In our proposed scheme, each trust evidence is missing, and default values are replaced by its field mean. Through Equations (5) to (9), we also normalize all the indicators and ensure that they increase monotonically.

3.2.1. Vehicle-Based Trust Evidence. We consider malicious vehicles have three types of attacks: generating fake messages, tampering with messages they received, and deliberately concealing messages about actual accidents. Vehicles receive multiple information about the same event sent by other CMVs and decide whether to forward the information. The number of information and correct information forwarded by the vehicle can reflect its identity. Two parameters TEV_1^i and TEV_2^i are proposed to represent the degree of selfishness and honesty of the vehicle i [16].

$$TEV_1^i = \frac{N_{send}^i(m)}{1/H\sum_{h=1}^{H} N_{send}^h(m)}, \tag{5}$$

$$TEV_2^i = \frac{\sum_{m=1}^{M} N_{honest}^i(m)}{\sum_{m=1}^{M} N_{send}^i(m)}, \tag{6}$$

where H is the number of neighbor vehicles that send messages about accident m, $N_{send}^i(m)$ is the number of messages that vehicle i sends to its neighbor's vehicle, M is the number of accidents that occurred on the roads. $\sum_{m=1}^{M} N_{honest}^i(m)$ is the total number of honest messages that vehicle i sends to its neighbors, and $\sum_{m=1}^{M} N_{send}^i(m)$ is the total number of messages that vehicle i sends.

3.2.2. Data-Based Trust Evidence. Data-based trust evaluations use the message to measure the vehicle's reliability. CHVs collect messages about each accident from different vehicles. Based on the spatial-temporal correlation, the quality of interactive information can be measured by its relevance to other information about the same accident. According to [25], we assume that the data obeys the normal distribution. The deviation between data and the average value reflects the reliability of the data. If the data is closer to the mean value, it will be more reliable than those far away. TEV_3 is defined to evaluate the trust degree of messages.

$$TEV_3^i = 1 - 2\int_{\mu}^{v_i^m} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx, \tag{7}$$

where v_i^m is the value of message transmitted by v_i.

3.2.3. Link-Based Trust Evidence. The link quality influences the accuracy of messages transmissions among vehicles. Considering attack patterns, such as Man-in-the-Middle attacks [26], attackers intercept normal network traffic data by attacking communication links and perform data tampering and sniffing. We measure the link quality from two aspects: link transmission delay and usage frequency [19].

$$TEV_4^i = 1 - \frac{l_{delay}(n_i)}{l_{delay}(n_i) + \sum_{j=1}^{H} l_{delay}(n_i)}, \tag{8}$$

$$TEV_5^i = \frac{l_{use}(n_i)}{l_{use}(n_i) + \sum_{j=1}^{H} l_{use}(n_i)}, \tag{9}$$

where TEV_4^i and TEV_5^i represent the link transmission delay and the link usage frequency, respectively, $l_{delay}(n_i)$ is the link transmission delay between vehicle n_i and its neighbor n_j, and $l_{use}(n_i)$ is the link usage of n_i.

After completing the trust evidence collection, we have five continuous variables. In order to further shorten the MTRF execution time and better meet the requirements of IoV delay sensitivity, we adopt the fuzzification method for $TEV_1^i \sim TEV_5^i$. Each data is converted into two-category variables $\{Low, High\}$ based on fuzzy rules to reduce the computational complexity and latency. If the data is less than *threshold*, the discrimination is *Low*; otherwise, the discrimination is *High*. Vehicles can further use the discretized trust evidence for trust degree classification.

3.3. Trust Evaluation Based on Random Forest. We adopt the Random Forest algorithm to evaluate vehicle reliability. Random Forest is an ensemble learning algorithm using bootstrap technology to extract a random sample set from the original sample set to construct a single decision [27]. Splitting nodes are selected to split at each node of the decision tree employing random feature subspace. Finally, these decision trees are combined to generate the final classification results through majority voting (bagging). RF synthesizes multiple deep decision trees that are trained on different parts of a training set to solve the overfitting problems by reducing variance instead of pruning processes. The details of RF are shown in Algorithm 1.

The CART is selected to generate trees because it uses GINI impurity metric to minimize classification error. S represents the training set with the size of N, which has class-labeled tuples. F represents the attribute set with the size of five. Y contains two types of target $\{High, Low\}$ and represents the trust degree of vehicles.

As previously described, vehicles have been grouped into several temporary clusters, and the formulas are shown in Equations (10) to (13). Let the total number of clusters as the P, and Num describe the number of vehicles in different clusters. For the $w-th$ cluster, each vehicle trains a CART tree, and the RF scale of cluster w is num_w. T^w represents the set of decision trees in the $w-th$ cluster. S_1^w and F_1^w are the training data extracted from S and the attribute set for the T_1^w, respectively.

$$Num = \{num_1, num_2, \cdots, num_P\}, \tag{10}$$

$$T^w = \left\{T_1^w, T_2^w, \cdots, T_{num_w}^w,\right\}, \tag{11}$$

$$S_1^w = \{s_{11}^w, s_{12}^w, \cdots, s_{1N}^w\}, \tag{12}$$

$$F_1^w = \{f_{11}^w, f_{12}^w, \cdots, s_{1m}^w\}. \tag{13}$$

It is important to note that each sample from S is extracted multiple times because N samples are randomly selected from the training set S with replacement. Each tree has a different training set, and a spot of identical samples appears in S_1^w. m is determined by the size of attribute set F. The training delay is not considered in our proposal because the classifiers are trained offline.

As demonstrated previously in Figure 3, once V_x discover the accident e occupied on the road, it will transmit $message_e$ to other vehicles in the same cluster. Then, $V_i, i = 1, 2, \cdots, x-1, x+1, \cdots num_w$ implements decision process based on F_i^w independently and forward the decision result $res_{x,i}^e$ to the CHV. Each $res_{x,i}^e, i = 1, 2, \cdots, num_w$ and $i \neq x$ is a target label that represents the trust evaluation result. Let RES_x^e represent the decision results made by vehicles in cluster w, and CHV integrates all $res_{x,i}^e$ to obtain RES_x^e which is transmitted to the corresponding RSU.

$$RES_x^e = \left\{res_{x,1}^e, \cdots, res_{x,(x-1)}^e, res_{x,(x+1)}^e, \cdots, res_{x,num_w}^e\right\}. \tag{14}$$

3.4. Trust Value Calculation and Update. Given the conceptual nature of "trust," both current and historical records can be used as benchmarks for trust values. Trust of vehicles can be evaluated based on historical data records, especially if there is no direct interaction between neighboring vehicles. The traditional RF algorithm finally uses the majority vote method to get the decision result. Since the RF training process requires a random selection of features and samples, it makes an inevitable training result unsatisfactory and leads to failure in the decision of vehicle trust. To solve the above issues, we make some improvements to traditional algorithms.

In our proposed algorithm, as shown in Figure 5, a sliding time window is used to store the trust characteristics of the vehicle nodes. The trust value calculation process consists of two parts: majority voting and trust accumulation.

As mentioned earlier, after CHV consolidates the decision results of vehicles in the cluster to obtain RES_x^e and uploads it to RSU, the RSU executes the majority voting process to obtain the classification result VR_x^t of the vehicle x at the current moment. In the follow-up process of trust accumulation, the sliding time window is adopted to perform a weighted summation of the trust values from time $t-(h-1)$ to t to obtain the final trust value of V_x according to Equation (15).

$$TV_x^t = \left(\frac{N_H^t}{N_H^t + N_L^t}\right)\left(\frac{1}{\sqrt[\delta]{N_L^t}}\right), \tag{15}$$

where N_H^t and N_L^t are the number of high and low classification results in VR, respectively. δ is a positive integer. The value of $\sqrt[\delta]{N_L^t}$ rises sharply with the increase in the number of low labels, reflecting the strict punishment characteristics for malicious vehicles.

Figure 6 is an effect diagram of Equation (15), where the horizontal axis is δ and the number of classified results as *Low*, and the vertical axis represents the trust value of the vehicle. Compared with the traditional linear relationship, the addition of the penalty factor δ makes the trust of the vehicle shows a rapid decline with the appearance of the

Input: Training set $S = \{s_1, s_2, \cdots, s_N\}$ with targets $Y = \{y_1, y_2, \cdots, y_N\}$
Attribute set $F = \{f_1, f_2, f_3, \cdots, f_M\}$
Output: A random Forest (S, Y, K)
For $k = 1 \longrightarrow K$ **do**
Bootstrapping:
For $k - th$ tree, N samples are randomly extracted from the training set S with replacement to constitute a new training set S_k with targets Y_k.
Specify a constant m that is far less than M, m features are randomly selected from the attribute set F as a new attribute set F_k.
Training:
Use S_k, Y_k, and F_k to perform CART procedure and train a classification T_k.
End for
Bagging:
The final classification is performed by majority vote which is based on the decision results generated by K trees.

Algorithm 1: Random forest algorithm.

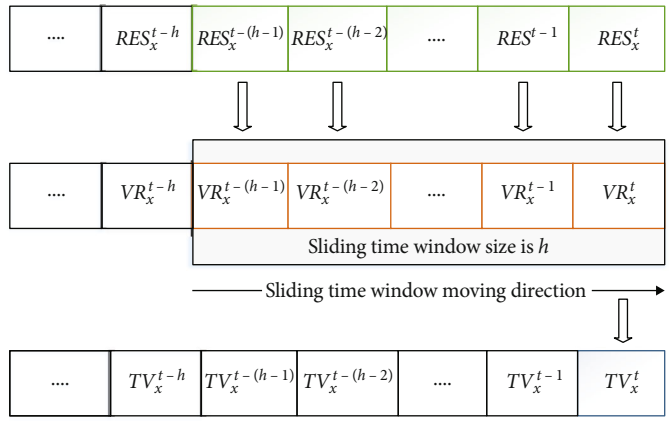


Figure 5: The sliding time window.

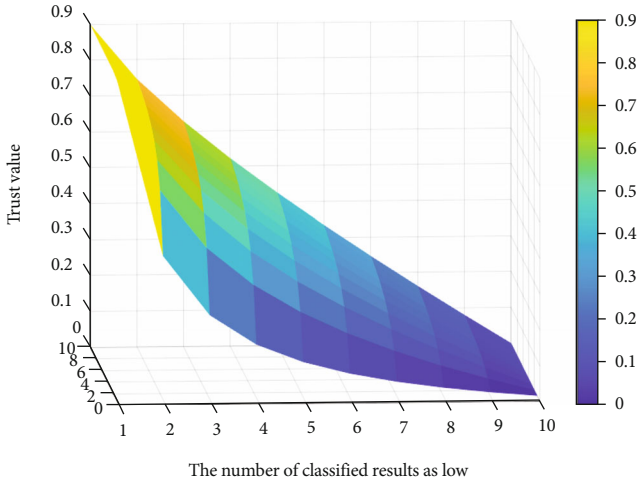


Figure 6: The performance under different δ.

malicious rating, reflecting the system's strict punishment characteristics for malicious vehicles. But at the same time, the penalty factor cannot be selected too small because the system needs to be fault-tolerant to decision failures caused by the randomness of the RF training process.

For MTRF, the computational complexity is mainly composed of RF decision process. Profit from the dynamic clustering mechanism, the evaluation process of the trust degree of the observed vehicles is restricted to a single cluster, which is uniformly managed by CHV. The out-of-cluster CMVs need not to participate in the decision-making process, resulting in lower computational complexity $O(\rho^2 N_{total}{}^2 \log(\rho N_{total})\sqrt{M})$, where $\rho \ll 1$ is the proportion of vehicles in the cluster to the total number of vehicles in the environment.

4. The Lightweight Cryptography Algorithm

With the help of MTRF, attacks by malicious vehicles broadcasting fake information among CMVs can be resisted effectively. However, this is not enough for a complex communication network such as IoV. IoV communication is carried out in an open wireless channel, where numerous types of adversarial behaviors exist. The communication processes between CMVs, CHV, and RSU lack protection mechanisms. Once attackers attack the above communication processes, information such as $res_{x,1}^e$ and RES_x^e are directly disclosed, which poses a severe threat to the effectiveness of MTRF.

Considering the limited computing, storage capacities of vehicles, and strict requirements for time delay for IoV, we propose a lightweight cryptography mechanism. The notations used are shown in Table 1.

Elliptic Curve Cryptography (ECC) is an asymmetric encryption algorithm based on the mathematical theory of elliptic curve. Compared with RSA, the ECC has the advantage of using shorter keys to achieve even higher security than RSA [28]. As shown in Table 1, F_p represents the finite field of a large prime number p, $E_p(a, b)$ is an elliptic curve defined by homogeneous Equation (16), where x, y, a, and b belong to F_p and are satisfied with Equation (17) [29]. And then $G(x_1, y_1)$ is assigned as the base point of $E_p(a, b)$ of which the order n is a large number. The random number N_* is less than n.

$$y^2 = x^3 + ax + b(\text{mod } p), \tag{16}$$

$$4a^3 + 27b^2(\text{mod } p) \neq 0. \tag{17}$$

The communication channels exposed to the environment are vulnerable to malicious attacks. Providing that only two communication processes are taken into consideration: CMV and CHV, CHV, and RSU. In addition, we assume that each type of road entity knows its identifier ID and private key PRK and generates its own public key PUK based on $PRK \times G$, which is shared in the communication network. The proposed cryptography mechanism is shown in Figure 7. If an CMV V_i makes a decision about the vehicle under observation and wants to transmit res_i^e to corresponding CHV, it will first encrypt res_i^e through hash function and self ID_{V_i} to generate $Msg1_i^e$ according to Equation (18) and then select a random number N_{V_i}. Finally, V_i transmits $\{N_{V_i}G, Msg1_i^e \oplus N_{V_i}PUK_{CHV}\}$ to CHV.

$$Msg1_i^e = res_i^e \| h(res_i^e \| ID_{V_i}). \tag{18}$$

After CHV received the information from V_i, it restores $Msg1_i^{e'}$ based on its PRK_{CHV} and then calculate $h(res_i^{e'} \| ID_{V_i})$. If $h(res_i^{e'} \| ID_{V_i}) = h(res_i^e \| ID_{V_i})$, res_i^e will be regarded as a complete and legal message. CHV then generates VR_x^t and $Msg2_x^e$ and transmits $\{N_{CHV}G, Msg2_x^e \oplus N_{V_i}PUK_{RSU}\}$ to RSU. The RSU performs the same steps to verify the received VR_x^t, calculates the TV_x^t, and returns it to the CHV to ensure the accuracy of TV_x^t.

We take the communication process of CHV-RSU as an example to verify the mechanism's effectiveness. CHV independently chooses the random number N_{CHV}, which is unknown to other road entities. It is almost impossible for an attacker to recover $Msg2_x^e$ from $\{N_{CHV}G, Msg2_x^e \oplus N_{V_i}PUK_{RSU}\}$, when only the RSU public key PUK_{RSU} is known. In addition, since the public key of the post-RSU is used to encrypt the information, the RSU is competent to use its private key to restore $Msg2_x^e$ based on Equation (19). To restore $Msg2_x^e$, an attacker must have G and $N_{CHV}G$ to solve for N_{CHV}, which is considered problematic.

Table 1: Main symbols used in the proposed cryptography mechanism.

Notation	Description
p	A large prime number
F_p	The finite field of p
$E_p(a, b)$	An elliptic curve defined by equation
G	The base point of $E_p(a, b)$
n	The order of G
ID_*	The identifier of traffic entity $*$
PRK_*,PUK_*	The private and public key of traffic entity $*$
N_*	The random number generated by $*$
$h(\cdot)$	The hash function
$\oplus$	The exclusive-OR operation
$\|\|$	The message concatenation operation

$$\begin{aligned} Msg2_x^{e'} &= \left(Msg2_x^{e'} \oplus N_{CHV}PUK_{RSU}\right) \oplus (PRK_{RSU}(N_{CHV}G)) \\ &= Msg2_x^{e'} \oplus N_{CHV}(PRK_{RSU}G) \oplus PRK_{RSU}(N_{CHV}G) \\ &= Msg2_x^{e'} = VR_x^{t'} \| h\left(VR_x^t \| ID_{CHV}\right). \end{aligned} \tag{19}$$

5. The Trust Sharing Algorithm

At present, most researchers use the central system controller to share the trust value of the vehicles across the cluster and adopt the soft handoff method. However, for the network with strong mobility and high delay sensitivity as IoV, it is confronted with the following three weaknesses:

(1) When the vehicle requires cluster switching, it must primarily establish a communication connection with the central controller and inform the target cluster. The trust value of the vehicle cannot be shared until the controller establishes communication with the target cluster, resulting in high communication delay and signaling overhead

(2) The location of the central controller is generally stationary, and the potency of trust sharing drops markedly as the distance between the vehicle and the controller increases, which imposes restrictions on the scalability of IoV.

(3) The employment of soft handoff makes the vehicle maintain the communication connection with the historical CHV before joining the new cluster, which is a waste of the communication resources of historical CHVs

In this paper, we combine trust information sharing with vehicle path prediction, so that trust information can be shared locally purposefully. Recently, reinforcement learning (RL) is developing rapidly and has a good application prospect in path prediction. RL is a principled mathematical

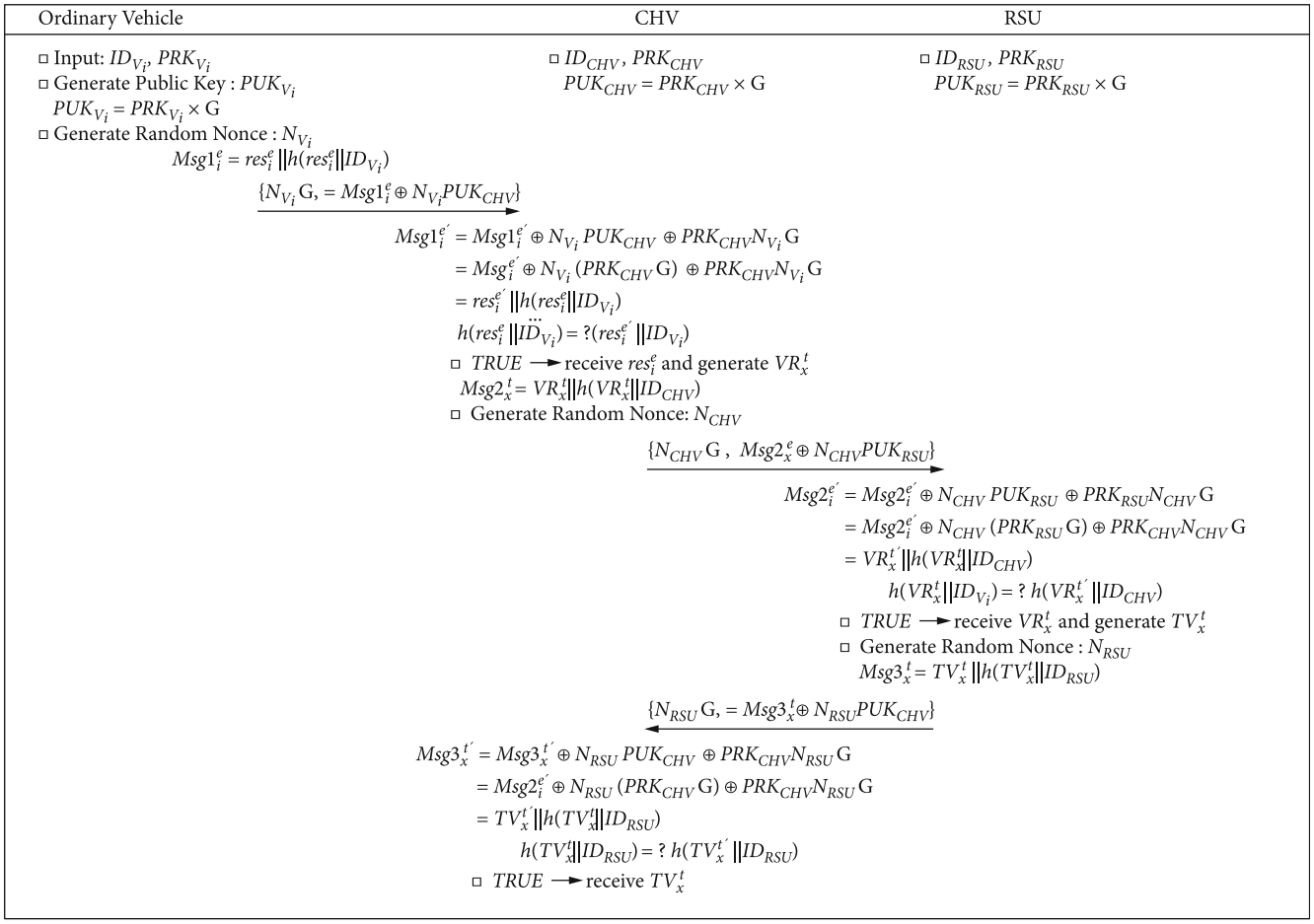


Figure 7: The cryptography mechanism based on ECC and hash function.

framework for experience-driven autonomous learning [30]. An agent learns how to maximize the benefits of a sequential decision problem by interacting with the environment. Formally, RL can be described as a Markov decision process (MDP), composed of a 5-dimension tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, R, \gamma)$, where $\mathcal{S}$ and $\mathcal{A}$ is the state and action set, respectively, $\mathcal{P}$ represents the state transition probability $\Pr(s_{t+1} \mid s_t, a_t)$, R stands for the expected reward set. At each time slot t, an agent observes state s_t and takes action a_t to make the interaction with environment. If the agent takes a_t, it will be transformed to a new state s_{t+1} and acquire a reward $r_t \in R$ based on the current state and the chosen action. The ultimate goal of RL agent is to find a policy π to maximize the cumulative reward $\mathbb{E}_\pi[\sum_{t=1}^{\infty} \gamma^{t-1} r_t]$, where γ is a discount factor and belongs to $[0, 1]$.

Q-learning is a widely used model-free RL algorithm that aims to find the Q-function of each state-action pair for the given policy, which is defined as

$$Q^\pi(s_t, a_t) = \mathbb{E}\left[\sum_{t'=1}^{\infty} r_{t'} \mid s_1 = s_t, a_1 = a_t\right], \tag{20}$$

where $Q^\pi(s_t, a_t)$ represents the cumulative reward when taking action a_t in state s_t and under the policy π. Q-learning updates the value function by time difference formula:

$$Q(s_t, a_t) \longleftarrow Q(s_t, a_t) + \alpha\left[r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)\right], \tag{21}$$

where α is the learning rate.

However, the traditional Q-learning algorithm learns the optimal policy by establishing and updating a Q-table, limiting the RL's scalability and ability to solve high-dimension problems. Deep Q-network, which is the combination of deep learning and Q-learning, is proposed to settle the above problems by using deep neural networks to approximate the value of the Q-table. The architecture of DQN is shown in fig.ẽffig:DQN, and after using DQN, Equation (22) can convert to

$$\theta_{t+1} \longleftarrow \theta_{t+1} + \alpha\left[r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, \mathrm{a}_{t+1}; \theta^-) - Q(s_t, a_t; \theta)\right] \times \nabla Q(s_t, a_t; \theta), \tag{22}$$

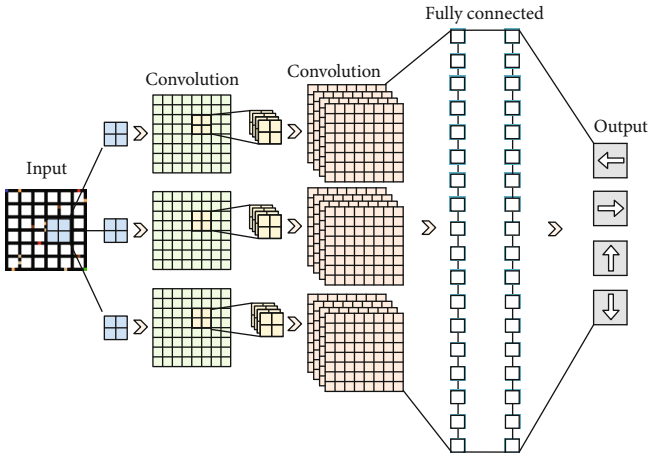


Figure 8: The architecture of DQN.

where $Q(s_t, a_t; \theta)$ and $Q(s_{t+1}, a_{t+1}; \theta^-)$ are the evaluation and target network, respectively, with different weight θ, θ^-, which is used to improve the training stability of DQN. It should be pointed out that the weight θ^- of target network is synchronized with θ periodically. Then we use the mean-square error to define the loss function. The network is trained by minimizing the loss, and finally $Q(s_t, a_t)$ is estimated.

$$L(\theta, \theta^-) = \mathbb{E}\left[\left(r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta^-) - Q(s_t, a_t; \theta)\right)^2\right]. \tag{23}$$

The formulation of path prediction mechanism is mainly divided into three parts: environment observation, action space design, and reward design.

5.1. Environment Observation. Environment observation is the input of the neural network. Whether the observation design is close to the natural environment information directly affects the availability of prediction results. So, the design of observation must accurately capture the characteristics of the application scenario. As for IoV, we take driving safety and driving efficiency as the focus to simulate the environment. The input of DQN is an RGB pixel image, which is shown in Figure 8, consisting of origin, terminus, and current point. To better simulate road conditions, the obstacles, flow, and accident points are also settled.

5.2. Action Space. In traffic path prediction algorithm, there are four types of actions in action space $\mathcal{A}$, $\mathcal{A} = \{up, down, left, right\}$. Each vehicle is an agent, which comprehensively considers the vehicle's current location and the surrounding environment. Vehicles choose different behaviors to interact with the environment and learn the best policy.

5.3. Reward Design. The core of RL is to learn unfamiliar scenes through interaction with the environment to obtain behavioral strategies to meet the set goals. In this process, the reward is the only feedback that an agent can obtain from the environment [31]. Rewards directly affect whether an agent can learn toward the desired goal and determine the model's effectiveness. Therefore, the design of rewards must fully reflect the expectation. For the consideration of driving safety and efficiency, the reward design of this mechanism focuses on four aspects: avoiding the section where traffic accidents are happening, avoiding the section with high vehicle density, avoiding the obstacle, and reaching the destination. The reward function is defined as folows:

$$reward_t = \begin{cases} r_{barrier}, & \text{if } crash \text{ the } barrier \\ r_{accident}, & \text{if } receive\ accident\ message \\ r_{flow}, & \text{if } flow > flow_{threshold} \\ r_{reach}, & \text{if } arrive\ at\ destination \\ 0, & otherwise \end{cases}, \tag{24}$$

where $flow_{threshold}$ is the maximum traffic flow that meets the normal driving speed of vehicles and according to the degree of need of different targets, $r_{barrier} < r_{accident} < r_{flow} < 0 < r_{reach}$.

Based on MTRF, we adopt the cryptography mechanism to prevent malicious attackers from destroying the communication connection between CHVS, CMVS, and RSUs, which effectively protects vehicle privacy information and decision results from being disclosed. After going through the trust management mechanism, the ultimate trust values of vehicles are stored in RSUs and propagated among RSUs according to the path prediction results. Considering that RSUs are mainly deployed by the government and have good authority and security, we assume that RSUs cannot be compromised by attackers. Under this assumption, when the RSU is not under attack, the private information of the vehicle is not easy to disclose.

6. Implementation and Performance Analysis

The simulation process consists of two parts: MTRF and DQN-based path prediction, and the parameters are shown in Table 2. As for the MTRF training process, each vehicle goes through a random selection of samples. Some samples are not involved in the training process for each vehicle's CART tree, called the out-of-bag samples. The system's accuracy can be evaluated by classifying the samples out of the bag by using RF [32]. We use the parameter *accuracy* to measure the performance of MTRF.

$$accuracy = 1 - OOB = 1 - \frac{N_{\text{out}-correct}}{N_{\text{out}}}, \tag{25}$$

where OOB represents the out-of-bag error, $N_{\text{out}-correct}$ is the number of correctly classified out-of-bag samples, and N_{out} is the total out-of-bag samples. In order to make the results more accurate and convincing, the decision-making process

Table 2: The core parameters.

Parameter	Parameter values
Simulation scene	Urban crossroad
Vehicle type	Honest + malicious vehicle
Attack behavior	Falsify and forward the decision results
Penalty factor δ	2
Extracted features rate	0.3
The number of vehicle in a single cluster	$1 \times$ CHV + $31 \times$ ordinary
The number of malicious vehicles in a cluster	3˜30
The probability of malicious vehicle attack	0.1˜1.0
The scale of the network	$7 \times 7 \sim 11 \times 11$
Environment size	25, 29, 33, 37, 41
Batch size	100
Learning rate	0.001
Epochs	8
$r_{barrier}$	-100
$r_{accident}$	-80
r_{flow}	-60, -40, -20
r_{reach}	+100

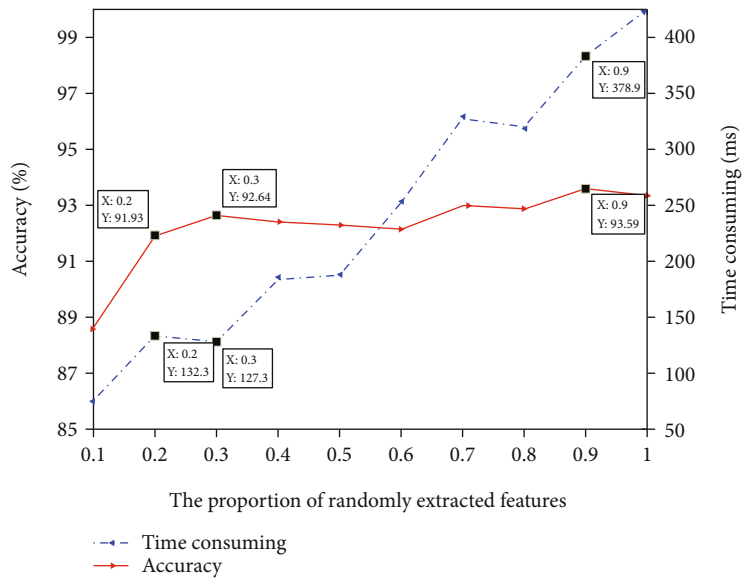


Figure 9: Accuracy and time-consuming under different extracted features rate.

of each vehicle is repeated 500 times to calculate the comprehensive value.

Figure 9 illustrates the accuracy and time-consuming with the variable of the proportion of randomly extracted features. It can be seen that as the proportion increases, the accuracy of MTRF first increases rapidly, and then gradually stabilizes, accompanied by a rapid increase in time consumption. This is because too small feature extraction rate results in incomplete growth of decision tree training and inability to accurately determine vehicle identity; excessive feature extraction rate results in complete growth of decision trees, and the forest composed of such trees is too capable to reflect the RF superiority. Time-consuming continues to increase as the complexity of the decision tree structure increases. Considering the accuracy of the system and the time-consuming comprehensively, the proportion of extracted features is 0.3, that is, when two features are selected, the accuracy and effectiveness of the system are both quite satisfactory. It should be noted that $featureextr\ actionrate \times M$ is a noninteger, and the rounding operation

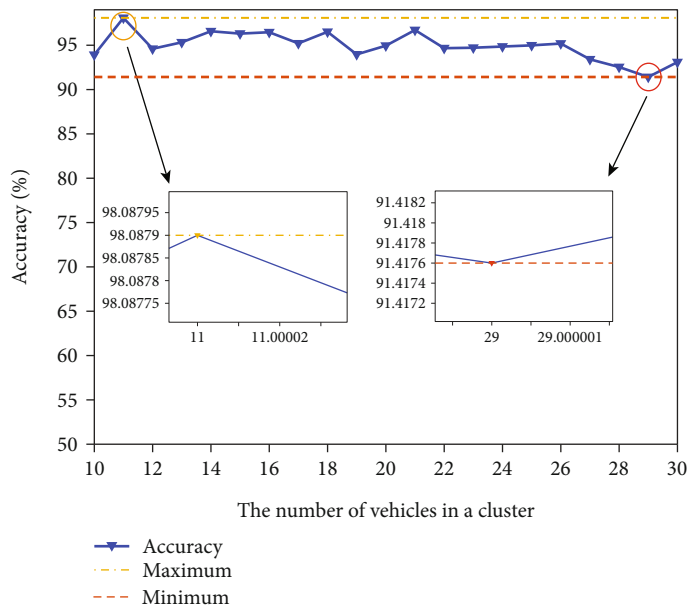


Figure 10: The performance under different number of vehicles in a cluster.

is adopted in the experiment, resulting in the same number of feature extraction, which causes the slow growth of the time-consuming curve at the early stage.

Figure 10 illustrates the accuracy of MTRF with a variable of the number of vehicles in a cluster, where the variable gradually increases from 10 to 30. The proportion of malicious vehicles remained 30%, and the probability of malicious vehicles attacking remained unchanged at 50%. It can be seen that as the number of vehicles in the cluster increases, the system's overall accuracy decreases slightly. When the number of vehicles is 11, the accuracy reaches the maximum 98.0879%, and when the number of vehicles is 29, the accuracy reached the lowest 91.4176%. The overall system accuracy remained above 90% with no apparent downward trend, which verified that the system we proposed could identify malicious vehicles, thereby improving the reliability of IoV. The random selection of samples and features during RF training results in slight fluctuations in the accuracy of MTRF, but within a reasonable range.

Figures 11 and 12 illustrate the accuracy of MTRF with the variables of the number of malicious vehicles in a cluster and the probability of launching a malicious attack.

Figure 11 shows the performance under different numbers of vehicles in a cluster. The four broken lines represent four different attack probabilities. It can be seen that when the attack probability is lower than 0.3, the accuracy of MTRF for malicious vehicle identification remains around 90% and does not change significantly with the increase of the number of malicious vehicles in the cluster. When the attack probability is higher than 0.5, the accuracy of MTRF decreases to a certain extent with the increase of the number of malicious vehicles in the cluster, but MTRF still maintains the accuracy of 77.8% until there are half of the malicious vehicles in the cluster.

Figure 12 shows the performance under different probabilities of launching attacks. The six broken lines represent six different numbers of malicious vehicles in a cluster. It can be seen that when the attack probability of malicious vehicles is 0.1, even if the malicious vehicles in the cluster increase to 27, the accuracy can be maintained above 92%. When the number of malicious vehicles in the cluster is less than 9, MTRF is almost not affected by the attack probability of malicious vehicles. Even if all attackers launch attacks in the cluster, MTRF can still maintain an accuracy rate above 83%. When the number of malicious vehicles is greater than 15, the accuracy of MTRF decreases as the probability of a malicious vehicle launching an attack increases, but the accuracy remains around 70% until the probability is 0.5.

The above results can prove that MTRF has a relatively superior effect. The RSU consolidates the results of all vehicles through the voting process, which is why only when both the number of malicious vehicles and the probability of launching attacks remains high, *accuracy* will be dramatically reduced. Malicious vehicles randomly choose whether to reverse the decision result according to the probability. When the proportion of malicious vehicles is less than 50% , it is difficult to affect the system classification results even if the attack is launched. When the proportion of malicious vehicles exceeds 50% and the probability of attack remains at a medium or low level, according to probability statistics, the probability of simultaneous attacks by vehicles is low so that the system can effectively resist attacks and the system accuracy is maintained at a high level.

We also compare the performance of MTRF with that of TECU proposed in [19] in the case of 30% malicious vehicles. As shown in Figure 13, under the same malicious vehicle number in a cluster and three different probabilities of malicious attack, the accuracy of MTRF is significantly better than TECU. In [19], the authors only realize the identification of malicious nodes but do not consider the attack behavior of malicious nodes. Once the node launches an attack, the system performance will decline sharply. Especially

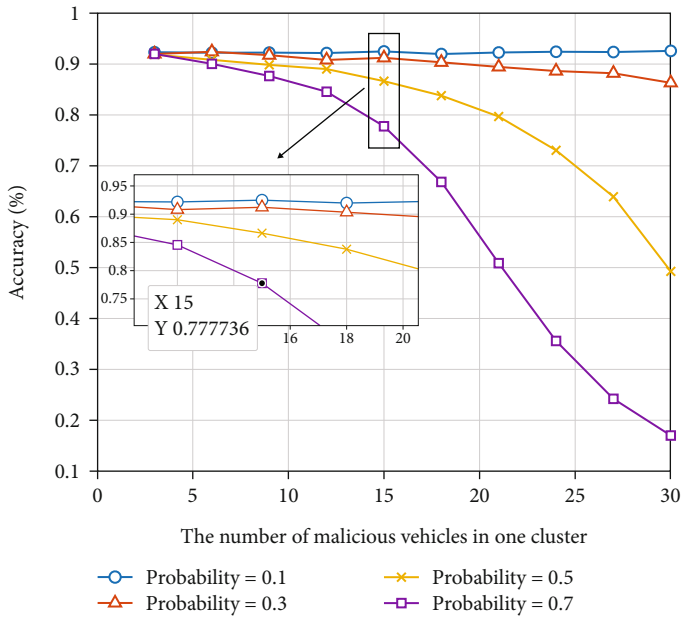


Figure 11: The performance under different numbers of malicious vehicles.

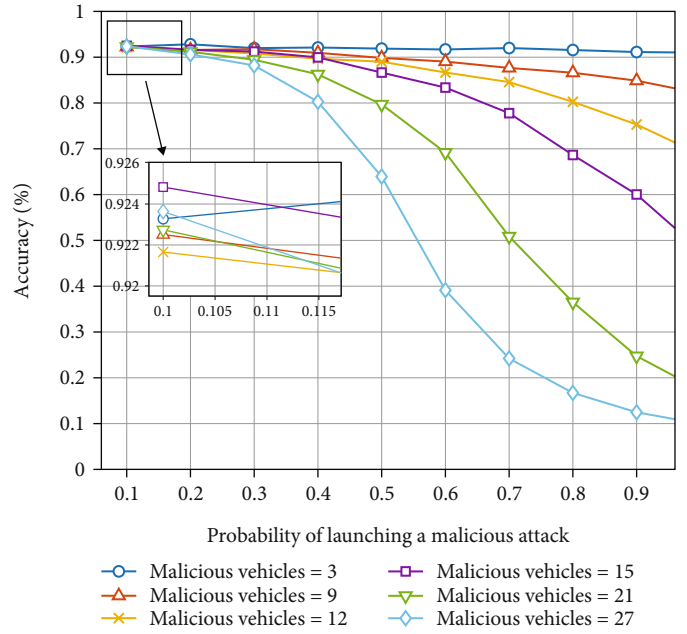


Figure 12: The performance under different probabilities of launching attacks.

when the probability of malicious attack is 0.5, the accuracy of TECU to identify malicious nodes is less than 80%. For MTRF, we use RF to make the vehicles in the cluster jointly identify malicious vehicles and weaken the attack effect of malicious nodes and keep the classification accuracy of MTRF above 90% even in the terrible environment. In addition, with the help of encryption algorithms, we further improve MTRF's ability to resist malicious vehicle attacks, thus maintaining superior performance.

The next portion is the simulation of the path prediction mechanism based on DQN. We use a 25×25 pixel image to simulate the traffic environment. As shown in Figure 14(a), the circumstance is composed of 25 intersections and 20 T-shaped intersections. Pixels have six corresponding RGB values. Black represents clear roads, white represents obstacles, blue and green represent the current location and destination of the vehicle, and red represents accidents. The other three colors represent the degree of congestion in the road, quantified by the traffic volume, and the degree of congestion increases as the color darkens.

There are two traffic accidents on the road. First of all, we simulate the path prediction with good traffic conditions.

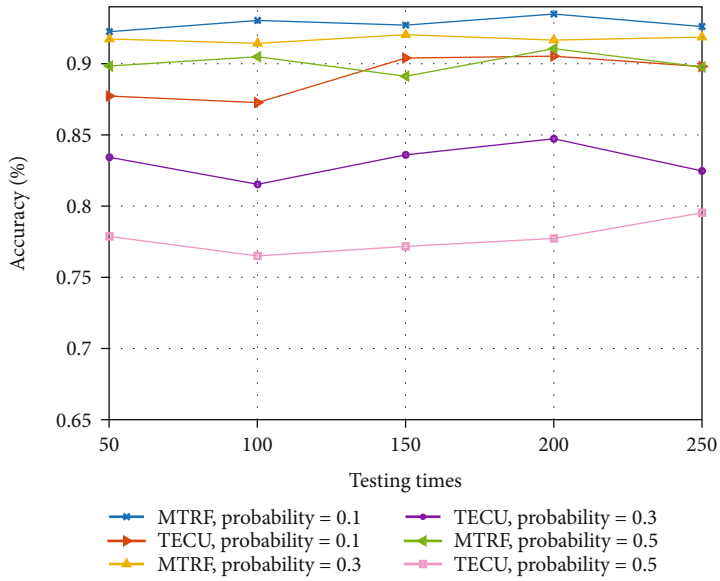
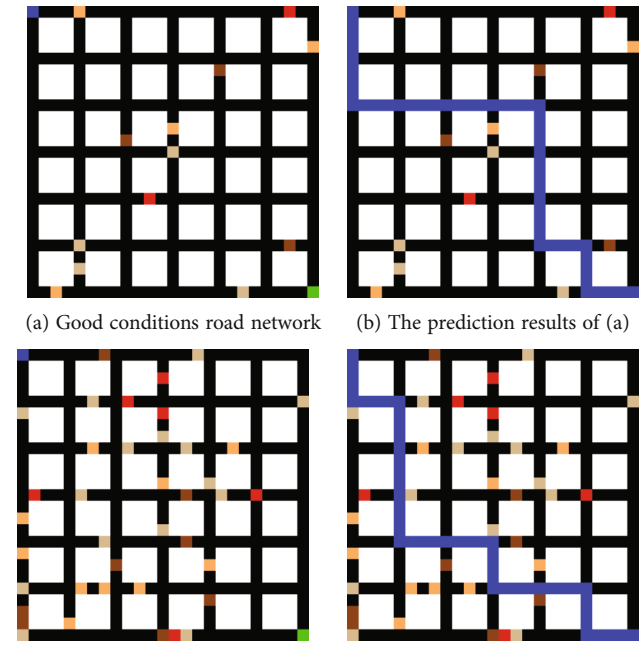


Figure 13: Comparison between MTRF and TECU.



(a) Good conditions road network (b) The prediction results of (a)

(c) Terrible conditions road network (d) The prediction results of (c)

Figure 14: DQN-based path planning environments and results.

There are two traffic accidents on the road, four sections with light congestion, four sections with moderate congestion, and three sections with severe congestion. It can be seen from the simulation results that the vehicle as an agent finds a path in the network that can avoid the above nodes and reach the destination safely. However, it is worth mentioning that there is more than one optimal path because the road conditions are relatively simple. Then we worsened the traffic situation, which consisted of six accident nodes, 15 lightly congested road sections, ten moderately congested road sections, and eight severely congested road sections. As shown in Figure 14(c), there is no perfect path in the current network, and the vehicle must experience congestion. However, according to our proposed algorithm, the vehicle chooses a path that only passes through a section of lightly congested traffic and obtains a better effect.

Figure 15 shows the average loss during the training process of the above two situations. It can be seen that there is a fast convergence rate in both scenarios, and the learning net would be desired. With the increase of the

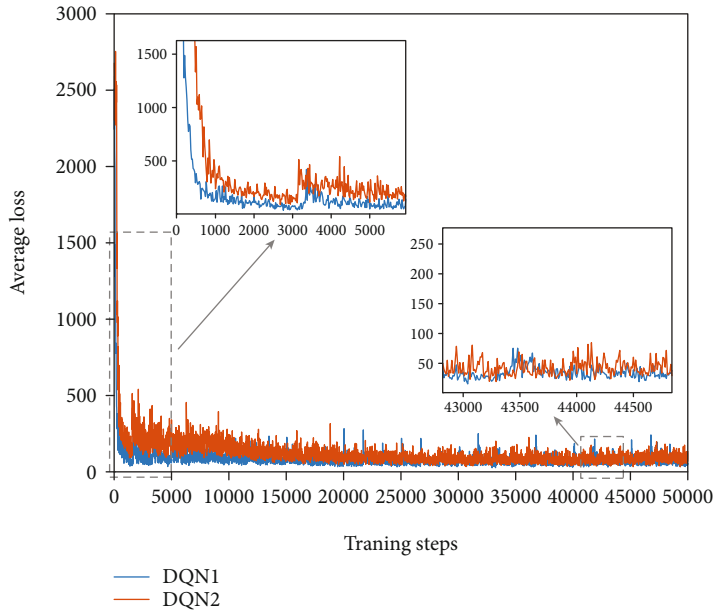


Figure 15: Average loss with the training time.

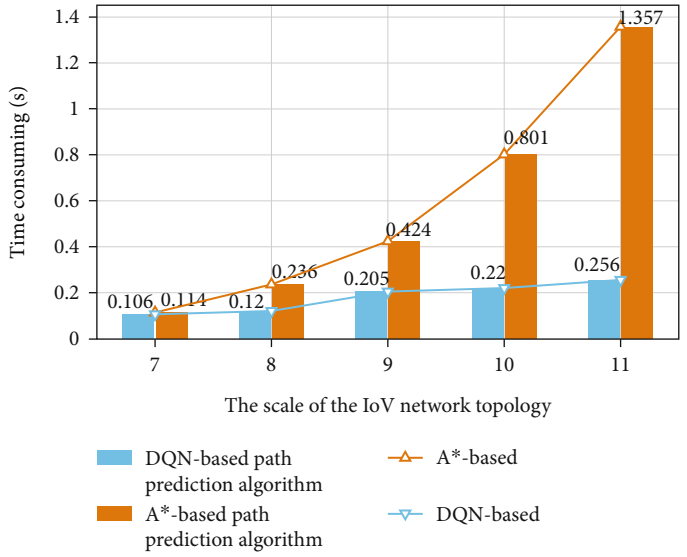


Figure 16: The time-consuming of two algorithms under different scales of IoV.

scene's complexity, the convergence speed does not significantly improve, with good generalization, which can cope with the scene of the IoV.

Finally, to more convincingly illustrate the superiority of the proposed path prediction mechanism, we compared the time consumed by the DQN-based path planning algorithm and the traditional A ∗-based algorithm under the same environmental conditions, and the results are shown in Figure 16. When the scale of IoV network is small, the difference between the time-consuming of the above algorithms is not apparent. With the increase of network topology scale, the delay and delay growth rate of the A ∗-based algorithm are much higher than those of our proposed algorithm, which showcases the superiority of our algorithm in terms of timeliness.

7. Conclusion

This paper proposed an efficient RF-based trust management mechanism MTRF tailored to the urban scenarios in IoV. We also proposed a trust-sharing mechanism based on path prediction using Deep Q-network. According to simulation results, we demonstrated the performance of our trust management scheme under different situations. In addition, we simulated the path prediction algorithms under different IoV network topology scales and different

traffic conditions to verify that the proposed mechanism can achieve good results and has good convergence performance. Compared with the traditional A ∗-based algorithm, the proposed algorithm can also highlight its better generalization and superiority in time-effectiveness. For the trust management mechanism, we mainly considered two types of attack modes, and lacked the considerations of other attack modes such as Sybil attack, which limited the defense capability of MTRF. In addition, the simulation of the urban road network environment took regular intersections as the basic unit, which simplified the complexity of roads to a certain extent. The defects mentioned above should be solved in future research.

Data Availability

The trust evidence data used to support the findings of this study came from scenario simulation and reasonable assumptions, which are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants 62102019, 61931001, and 61871023 and in part by the Fundamental Research Funds for the Central Universities under Grants 2021RC222, 2021RC225, and 2019JBZ001.

References

[1] L. Chettri and R. Bera, "A comprehensive survey on Internet of Things (IoT) toward 5G wireless systems," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 16–32, 2020.

[2] B. Ji, X. Zhang, S. Mumtaz et al., "Survey on the internet of vehicles: network architectures and applications," *IEEE Communications Standards Magazine*, vol. 4, no. 1, pp. 34–41, 2020.

[3] H. Hasrouny, A. E. Samhat, C. Bassil, and A. Laouiti, "VANet security challenges and solutions: a survey," *Vehicular Communications*, vol. 7, pp. 7–20, 2017.

[4] F. Yang, J. Li, T. Lei, and S. Wang, "Architecture and key technologies for internet of vehicles: a survey," *Journal of Communications and Information Networks*, vol. 2, no. 2, pp. 1–17, 2017.

[5] C. R. Storck and F. Duarte-Figueiredo, "A survey of 5G technology evolution, standards, and infrastructure associated with vehicle-toeverything communications by internet of vehicles," *IEEE Access*, vol. 8, pp. 117593–117614, 2020.

[6] S. S. Manvi and S. Tangade, "A survey on authentication schemes in vanets for secured communication," *Vehicular Communications*, vol. 9, pp. 19–30, 2017.

[7] M. Blaze, J. Feigenbaum, and J. Lacy, "Decentralized trust management," in *IEEE Symposium on Security and Privacy*, Oakland, CA, USA, 1996.

[8] J. Y. Choi, M. Jakobsson, and S. Wetzel, "Balancing auditability and privacy in vehicular networks," in *ACM International Workshop on Quality of Service and Security in Wireless and Mobile Networks*, Montreal, CA, 2005.

[9] H. Vasudev, V. Deshpande, D. Das, and S. K. Das, "A lightweight mutual authentication protocol for v2v communication in internet of vehicles," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 6709–6717, 2020.

[10] F. Ahmad, F. Kurugollu, A. Adnane, R. Hussain, and F. Hussain, "MARINE: man-in-the-middle attack resistant trust model in connected vehicles," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3310–3322, 2020.

[11] P. Wang, C.-M. Chen, S. Kumari, M. Shojafar, R. Tafazolli, and Y.-N. Liu, "HDMA: hybrid d2d message authentication scheme for 5G-enabled vanets," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 8, pp. 5071–5080, 2021.

[12] H. Hu, R. Lu, Z. Zhang, and J. Shao, "REPLACE: a reliable trust-based platoon service recommendation scheme in vanet," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 2, pp. 1786–1797, 2017.

[13] S. Ahmed, S. Al-Rubeaai, and K. Tepe, "Novel trust framework for vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 10, pp. 9498–9511, 2017.

[14] X. Li, J. Liu, X. Li, and W. Sun, "Rgte: a reputation-based global trust establishment in vanets," in *2013 5th International Conference on Intelligent Networking and Collaborative Systems*, pp. 210–214, Xi'an, China, 2013.

[15] S. A. Soleymani, A. H. Abdullah, M. Zareei et al., "A secure trust model based on fuzzy logic in vehicular ad hoc networks with fog computing," *IEEE Access*, vol. 5, pp. 15619–15629, 2017.

[16] S. Guleng, C. Wu, X. Chen, X. Wang, T. Yoshinaga, and Y. Ji, "Decentralized trust evaluation in vehicular internet of things," *IEEE Access*, vol. 7, pp. 15980–15988, 2019.

[17] T. Halabi and M. Zulkernine, "Trust-based cooperative game model for secure collaboration in the internet of vehicles," in *ICC 2019 – 2019 IEEE International Conference on Communications (ICC)*, pp. 1–6, Shanghai, China, 2019.

[18] F. Tang, B. Mao, N. Kato, and G. Gui, "Comprehensive survey on machine learning in vehicular network: technology, applications and challenges," *IEEE Communications Surveys Tutorials*, vol. 23, no. 3, pp. 2027–2057, 2021.

[19] J. Jiang, X. Zhu, G. Han, M. Guizani, and L. Shu, "A dynamic trust evaluation and update mechanism based on c4.5 decision tree in underwater wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 8, pp. 9031–9040, 2020.

[20] P. K. Singh, R. Singh, S. K. Nandi, K. Z. Ghafoor, D. B. Rawat, and S. Nandi, "Blockchain-based adaptive trust management in internet of vehicles using smart contract," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3616–3630, 2021.

[21] Y.-T. Yang, L.-D. Chou, C.-W. Tseng, F.-H. Tseng, and C.-C. Liu, "Blockchain-based traffic event validation and trust verification for vanets," *IEEE Access*, vol. 7, pp. 30868–30877, 2019.

[22] C. A. Kerrache, C. T. Calafate, J.-C. Cano, N. Lagraa, and P. Manzoni, "Trust management for vehicular networks: an adversary-oriented overview," *IEEE Access*, vol. 4, pp. 9293–9307, 2016.

[23] Q. Han, Q. He, L. Zeng, L. Ye, and F. Li, "A bus oriented coordination method for intra-cluster bsm transmission," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 3358–3363, Maui, HI, USA, 2018.

[24] H. Xiao, W. Zhang, W. Li, A. T. Chronopoulos, and Z. Zhang, "Joint clustering and blockchain for real-time information security transmission at the crossroads in c-v2x networks," *IEEE Internet of Things Journal*, vol. 8, no. 18, pp. 13926–13938, 2021.

[25] J. Jiang, G. Han, F. Wang, L. Shu, and M. Guizani, "An efficient distributed trust model for wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 5, pp. 1228–1237, 2015.

[26] M. Conti, N. Dragoni, and V. Lesyk, "A survey of man in the middle attacks," *IEEE Communications Surveys Tutorials*, vol. 18, no. 3, pp. 2027–2051, 2016.

[27] A. Jahangiri and H. A. Rakha, "Applying machine learning techniques to transportation mode recognition using mobile phone sensor data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2406–2417, 2015.

[28] D. He and S. Zeadally, "An analysis of RFID authentication schemes for internet of things in healthcare environment using elliptic curve cryptography," *IEEE Internet of Things Journal*, vol. 2, no. 1, pp. 72–83, 2015.

[29] D. He, S. Zeadally, B. Xu, and X. Huang, "An efficient Identity-Based conditional privacy-preserving authentication scheme for vehicular ad hoc networks," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2681–2691, 2015.

[30] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: a brief survey," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, 2017.

[31] L. Lv, S. Zhang, D. Ding, and Y. Wang, "Path planning via an improved dqn-based learning policy," *IEEE Access*, vol. 7, pp. 67319–67330, 2019.

[32] F. Valle, S. Cespedes, and A. S. Hafid, "Automated decision system to exploit network diversity for connected vehicles," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 1, pp. 858–871, 2021.