A Trust Modeling Framework for Message Propagation and Evaluation in VANETs

Chen Chen Jie Zhang[†] Robin Cohen Pin-Han Ho[‡]
David R. Cheriton School of Computer Science, University of Waterloo, Canada
[†]School of Computer Engineering, Nanyang Technological University, Singapore
[‡] Department of Electrical and Computer Engineering, University of Waterloo, Canada
{c32chen}@uwaterloo.ca

***Abstract*—In this paper, we present a trust-based message propagation and evaluation framework in vehicular ad-hoc networks where peers share information regarding road condition or safety and others provide opinions about whether the information can be trusted. More specifically, our trust-based message propagation model collects and propagates peers' opinions in an efficient, secure and scalable way by dynamically controlling information dissemination. The trust-based message evaluation model allows peers to evaluate the information in a distributed and collaborative fashion by taking into account others' opinions. Experimental results demonstrate that our proposed framework promotes network scalability and system effectiveness in information evaluation under the pervasive presence of false information, which are the two essentially important factors for the popularization of vehicular networks.**

I. Introduction

With the advance and wide deployment of wireless communication technologies, vehicle manufactures and research academia are heavily engaged in the blueprint of future vehicular ad-hoc networks (VANETs). Peers (vehicles) in a VANET communicate with each other by sharing road condition and safety information, to enhance passenger and road safety and to effectively route traffic through dense urban areas. Tremendous effort has been spent on the development of life-critical or road condition related systems, such as traffic view systems [1], safety message sharing [2], cooperative collision avoidance [3], and secure crash reporting [4]. These systems focus mainly on ensuring a reliable delivery of messages among peers. As a result, less focus has been placed on evaluating the quality of information that is sent by peers, in order to cope with reports from malicious peers which may compromise the network, without the assumption of a pervasively available infrastructure such as an online central authority or road side units. In addition, little concern has been focused on the design of a control mechanism where upon detection of false information, it should be immediately controlled to minimize its further negative effect on other peers in the network.

In this paper, we propose a trust-based message propagation and evaluation framework to support the effective evaluation of information sent by peers and the immediate control of false information in a VANET. More specifically, our trust-based message propagation collects peers' trust opinions about a message sent by a peer (message sender) during the propagation of the message. We improve on an existing cluster-based data routing mechanism by employing a secure and efficient identity-based aggregation scheme for the aggregation and propagation of the sender's message and the trust opinions. These trust opinions weighted by the trustworthiness of the peers modeled using a combination of role-based and experience-based trust metrics are used by cluster leaders to compute a majority opinion about the sender's message, in order to proactively detect false information. Malicious messages are dropped and controlled to a local minimum without further affecting other peers. Our trust-based message evaluation allows each peer to evaluate the trustworthiness of the message by also taking into account other peers' trust opinions about the message and the peer-to-peer trust of these peers. The result of the evaluation derives an effective action decision for the peer.

We evaluate our framework in simulations of real life traffic scenarios by employing real maps with vehicle entities following traffic rules and road limits. Some entities involved in the simulations are possibly malicious and may send false information to mislead others or spread spam messages to jam the network. Experimental results demonstrate that our framework significantly improves network scalability by reducing the utilization of wireless bandwidth caused by a large number of malicious messages. Our system is also demonstrated to be effective in mitigating against malicious messages and protecting peers from being affected. Thus, our framework is particularly valuable in the deployment of VANETs by archiving a high level of scalability and effectiveness.

II. Overview

Three types of messages are generated in our system: 1) Sender message: $M = [event, confidence, time, location]$. $confidence \in [0, 1]$ provides flexibility in reporting an event, $time \in N$ is a positive integer and $location \in N \times N$ is a geographical coordinate, both being available from an equipped GPS device; 2) Trust opinion: $O = [reaction, confidence]$, where $reaction \in \{trust, \neg trust\}$ and $confidence \in [0, 1]$. A trust opinion is a message provided by a peer that serves as an evaluation of the sender message; 3) Aggregated message:

$A = [M, O_1, \ldots, O_n]$, a combination of a sender message and a list of trust opinions.

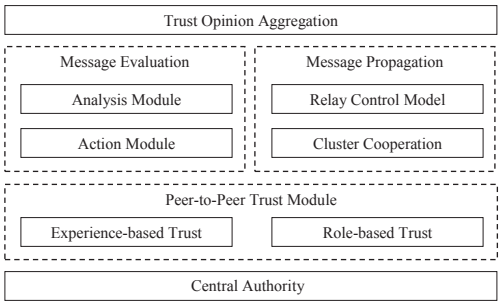

Fig. 1. Design of Framework

Figure 1 illustrates the modular design of our trust-based framework composed of several major components. Message evaluation contains two modules: analysis module and action module. The analysis module generates trust opinions. It analyzes a sender message's validity, correctness and accuracy based on a peer's local knowledge, and attempts to provide a trust opinion of either "trust" or "¬trust". One important design principle is that the trust opinion should always be generated before any disclosure of the existing trust opinions in the aggregated message. In other words, the generation of the trust opinion is purely based on the peer's local knowledge, such as direct observations. By doing so, we are capable of coping with gambling peers who give trust opinions by strategically guessing the message trustworthiness from others' trust opinions so as to quickly and maliciously increase their trust. If a trust opinion can be provided, it is broadcasted and appended to the sender message. The action module is where a local decision is made. It derives a local action using a trust-based computation model that will be described in Section III-C.

Message propagation consists of two components: cluster cooperation and the relay control model. Based on a cluster-based routing mechanism, the cluster cooperation serves as the foundation for message propagation and trust opinion aggregation. The relay control model works as a filter that controls the relay of messages. The trust opinion aggregation scheme ensures that message evaluation and propagation can be done with little interference on each other. It provides high flexibility that during message propagation, trust opinions can be aggregated in a secure, scalable and efficient fashion.

Peer-to-peer trust module manages the trustworthiness of peers. Motivated by the approach of [5], we employ both *role-based trust* and *experience-based trust*. A minority of vehicles, such as police cars, are assigned by a specific role and a specific role trust value. For other vehicles, they are associated with experience-based trust. Each peer maintains experience-based trust for other peers. The offline central authority assigns roles and updates role-based trust, collects distributed experience-based trust from peers, and praises or punishes peers accordingly. We provide detailed descriptions of these major components in the following sections.

III. Trust Opinion Aggregation and Propagation

In this section, we describe how trust opinions from peers can be effectively aggregated and propagated in the VANET, and also demonstrate how they help a single peer to derive a local action decision.

A. Cluster-based Aggregation

Message relay between each pair of neighboring peers in VANETs often results in wireless channel congestion. To achieve scalable trust opinion aggregation, we rely on a cluster-based data routing mechanism. A number of cluster-based routing protocols have been proposed to achieve scalability for vehicle-to-vehicle messaging [6]. By grouping peers into multiple clusters, the system becomes scalable by having message relay done between cluster leaders instead of between two neighboring peers. We extend the existing cluster-based routing protocols in two aspects. First, trust opinions from members in the cluster are aggregated and relayed along with the message itself so that the number of messages passed between peers is significantly decreased. Second, we employ the majority opinion computed from trust opinions as the decision of the relay control model, which further increases the scalability of the network by reducing the network bandwidth utilized by malicious messages.

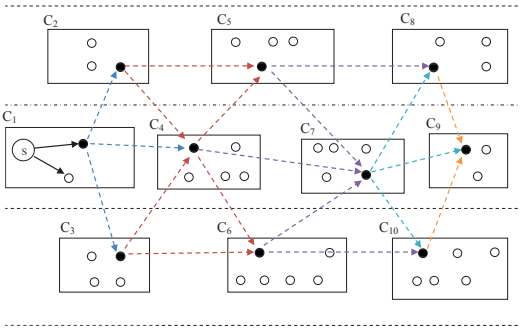

Fig. 2. Cluster-based Message Propagation

As demonstrated by an example shown in Figure 2, vehicles (peers) are geographically grouped into 10 clusters, i.e. from C_1 to C_{10}. For each cluster C_i, a vehicle is randomly chosen from all cluster members (the white nodes) as the cluster leader L_i (the black nodes). Our scheme requires that the cooperation among neighboring cluster leaders is pre-established to help build an intra-cluster link topology (the graph with dashed arrows connecting neighboring black peers) so that messages can be relayed from one cluster to another. Sender s in cluster C_1 broadcasts a message M to its members who will provide their trust opinions O_i immediately afterwards. After that, the cluster leader L_1 collects O_i and aggregates them into the aggregated message A. L_1 sends A to the next hop clusters C_2, C_3 and C_4. Upon reception of A, the cluster leader (e.g. L_4 here) broadcasts A to its cluster members, collects their trust opinions (if any), aggregates them together with the existing A into a new aggregated message A', and computes a relay

decision about whether to relay A' to the next hop cluster C_5, C_6 and C_7.

To implement our message aggregation protocol, a secure and efficient aggregation scheme is required. The secure aggregation would require a signature along with each message being sent, which brings two advantages. First, messages cannot be maliciously modified without being detected. Second, once messages are signed, peers cannot deny the fact that the messages are sent by them. Aggregation should also be efficient, otherwise it would render the system unscalable. We propose an aggregation scheme that extends the identity-based aggregate signature algorithm [7]. For example, the sender s sends a message $M_0 = [M, \mathrm{ID}_0, G_0]$ where ID_0 is the sender's identity and G_0 is the signature of M_0. Each peer i provides a trust opinion $M_i = [M, O_i, \mathrm{ID}_i, G_i]$ for $i \in [1, n]$. An aggregator computes $G' = \sum_{i=0}^{n} G_i$, and generates the aggregated message $A = [M, O_1, \ldots, O_n, \mathrm{ID}_0, \mathrm{ID}_1, \ldots, \mathrm{ID}_n, G']$. The summation of G_i is implemented over bilinear groups constructed by the modified Weil pairing on elliptic curves [8].

B. Relay Control Model

While traditional routing algorithms [9] in vehicular networks use "time-to-live" or "hop-to-live" as a relay decision, our decision is determined by the majority opinion: a message trusted by the majority should be relayed; otherwise it is to be dropped. Formally, let P be a set of peers whose trust opinions are "trust", $P = \{i|\ \mathrm{ID}_i \in A \text{ and } O_i = [trust, c_i] \in A\}$, and P' be a set of peers whose trust opinions are "¬trust", $P' = \{i|\ \mathrm{ID}_i \in A \text{ and } O_i = [\neg trust, c_i] \in A\}$. A relayer (cluster leader) L computes the weight of "trust" and "¬trust" opinions respectively as:

$$W_{trust} = \sum_{i \in P} c_i T_i, \quad W_{\neg trust} = \sum_{i \in P'} c_i T_i \tag{1}$$

and $T_i \geq \tau$, where τ is a trust threshold set by L, $c_i \in [0, 1]$ is the confidence given by peer i, and T_i is the peer-to-peer trust of peer i. We will introduce the peer-to-peer trust in Section IV. Messages can be relayed only if

$$\frac{W_{trust}}{W_{trust} + W_{\neg trust}} > 1 - \varepsilon \tag{2}$$

where $\varepsilon \in [0, 1]$ is a threshold set by the system to denote the maximum error rate allowed. ε is embedded in the protocol and can be adaptive to the current environment, situations and data types. For example, for more critical messages, such as car accidents, a lower error rate is appreciated.

Trustworthiness of messages ages with the time and distance. The longer time elapses and the further away the event incurs, the less accurate and reliable the data becomes. We use a mapping function $f_{max} : \Lambda \times \Theta \rightarrow M_t \times M_d$ which maps the sender role Λ and the event Θ to the maximum time-to-live M_t and largest propagation distance M_d. We define such a mapping function because it is reasonable to set different thresholds for multiple types of messages and for different types of senders. Take the distance M_d for an example. A piece of weather information can have a propagation area of 10 square miles while a life-critical message, e.g. "sudden brake" may only be useful within a distance of 200 meters. Similarly, the message from an authority role should propagate as far as possible. In short, the relay decision is also based on the following parameters: M_d, the maximum propagation distance; M_t, the longest time to live; Δd, the distance between current location and event location; and Δt, the time that has elapsed since the event occurs. The relayer's relay control decisions take four steps: 1) verify the aggregated message A; in case verification fails, drop A; 2) compute $\Delta d, M_d, \Delta t, M_t$, if $\Delta d > M_d$ or $\Delta t > M_t$, drop A; 3) compute the weight of opinion; drop A if the majority distrusts A (see Equation 2); 4) generate a new aggregated message A' by attaching new trust opinions of cluster members and relay A' to the next hop clusters.

Grouping peers into clusters and relaying messages between cluster leaders increase much the scalability of the system. Our relay control model further proactively detects malicious messages during information dissemination. Malicious data is therefore dropped and controlled to a local minimum without further affecting other peers. We will demonstrate such an important feature of our framework in vehicular networks in Section V. An aggregated message propagated through our message propagation scheme is then used by the action module to derive an action decision for a peer.

C. Action Module

The action module derives a local decision for a peer to take an action towards a sender message from trust opinions for the message. Specifically, the aggregated trustworthiness of the message is computed and mapped to an action set $\{follow, \neg follow\}$. Let A denote the aggregated message, s denote the original sender, P denote the peers who contribute trust opinions of "trust", and P' denote the peers with opinions of "distrust". Let T_A denote the aggregated trustworthiness of the message A. The action module of peer p computes:

$$T_A = \frac{c_s + \sum_{i \in P} c_i \ - \sum_{i \in P'} c_i}{1 + |P| + |P'|} \tag{3}$$

where $c_s \in [0, 1]$ is the sender's confidence in the sender message, $c_i \in [0, 1]$ is the confidence in the trust opinion given by peer i, and $T_A \in (-1, 1]$. T_A approaches -1 when $P = \emptyset$, $c_i = 1$ for $i \in P'$ and $|P'|$ is large, meaning that the message is fully distrusted. $T_A = 1$ when we have $c_s = c_i = 1$ for $i \in P$ and $P' = \emptyset$, which indicates that the message is fully trusted by the peer.

Considering that the sender is a different role from those who provide trust opinions, we employ a sender weight factor $\gamma > 0$ that determines how much weight is placed on the sender. The value of γ can be customized by each peer in the network. Setting γ to a larger value indicates that the peer places more trust on the sender. Considering that the peer's honesty varies, we also employ the peer-to-peer trust module. Each peer i is associated with a trust metric $T_i \in [0, 1]$. We combine the sender weight and the trustworthiness of each

peer into the computation for the aggregated trustworthiness of the message A as follows:

$$T_A = \frac{\gamma c_s T_s + \sum_{i \in P} c_i T_i - \sum_{i \in P'} c_i T_i}{\gamma T_s + \sum_{i \in P} T_i + \sum_{i \in P'} T_i} \tag{4}$$

and $T_i \geq \tau$, where $\tau \in [0, 1]$ is the trust threshold customized by each peer p. The trust threshold helps filter trust opinions from those peers that are not highly trusted. τ can be set to a higher value close to 1 so that only trust opinions from highly trusted peers will be used. In practice, the value of τ should be determined by the availability of trust opinions. For example, τ can be set higher when a larger number of trust opinions are available.

The action module implements a mapping $f_{action} : T_A \rightarrow \{follow, \neg follow\}$ that maps the trustworthiness of the message to an action:

$$f_{action} = \begin{cases} follow & \text{if } T_A \geq \varphi, \\ \neg follow & \text{otherwise.} \end{cases} \tag{5}$$

where $\varphi \in [-1, 1]$ is the action threshold. The value of φ can be personalized by each peer: a higher action threshold indicates the peer is more "cautious" of following other peers' advice and vice versa. Under the special situation where the traffic is extremely sparse, both P and P' may be $\emptyset$ and the message only contains the sender's identity. If we simply compute the aggregated trustworthiness using Equation 4, which becomes $T_A = (\gamma c_s T_s + 0 + 0)/(\gamma T_s + 0 + 0) = c_s$, the trust of the sender is eliminated and thus not considered. Therefore, along with the previous requirement in Equation 5 that $T_A = c_s \geq \varphi$, we further require that a peer follows the message only if $T_s \geq \tau$.

IV. Peer-to-Peer Trust Module

In our system, each peer's trust is evaluated by a trust metric: either role-based trust or experience-based trust. Let $T_i \in [0, 1]$ denote the peer-to-peer trust of peer i, we have

$$T_i = \begin{cases} T_i^r & \text{if peer } i \text{ has a role,} \\ f(T_{i,p}^e) & \text{otherwise.} \end{cases} \tag{6}$$

where $T_i^r \in [0, 1]$ is the role-based trust of peer i, and $T_{i,p}^e \in [-1, 1]$ is the experience-based trust of peer i from peer p's perspective. We map the value of T^e to the same range of T^r by employing a mapping function, for example $f(x) = (x + 1)/2$.

A. Role-based Trust

It is known that although most vehicles are for personal purposes, a small number of entities have their specific responsibilities in the traffic system, e.g. police cars. Roles are assigned to them and it is reasonable to assign multiple levels of trust to different roles. The underlying assumption is that vehicles of the same role would behave in a similar way so that any third party can estimate their trust levels before any interaction happens. The roles and role-based trust values in our system are fixed by the offline central authority. To demonstrate the utilization of role-based peer trust, we define three different roles, from the highest to the lowest trust: i) authority, such as police cars, traffic controllers, and road-side units which serve as part of road infrastructure, and so on; ii) public services, which could be ambulance, fire truck, school bus, public transits, road maintenance cars, etc; iii) professional cars, e.g. driver training vehicles, cars whose drivers have more than ten years of safe driving experience, etc.

We denote the role-based trust of peer i as T_i^r, where $T^r : \text{ID} \rightarrow [0, 1]$; 1 means absolute trust and 0 represents absolute distrust. The vehicle identity can be mapped to its role and then the role-based trust value. In practice, vehicles periodically download from the offline central authority an up-to-date list of roles, each with a list of vehicle identities.

B. Experience-based Trust

For most of the peers who do not have a role, we use the experience-based peer trust to dynamically reflect a peer's trustworthiness in the system. The behavior of a peer is evaluated by other peers, each of whom maintains trustworthiness for a list of peers in the system. The list of trust is preserved in peer's local repository.

We denote the peer i's experience-based trust from p's perspective as $T_{i,p}^e$, whose value is in the range of $[-1, 1]$. We simplify the notation of $T_{i,p}^e$ as T in the following formalization. Adapted from [10], if i's trust opinion leads to a correct decision of p, peer p increases the trust of i by

$$T \leftarrow \begin{cases} \lambda^t (1 - c\alpha) T + c\alpha & \text{if } T \geq 0 \\ \lambda^{-t} (1 + c\alpha) T + c\alpha & \text{if } T < 0 \end{cases} \tag{7}$$

otherwise, decreases T by

$$T \leftarrow \begin{cases} \lambda^t (1 + c\beta) T - c\beta & \text{if } T \geq 0 \\ \lambda^{-t} (1 - c\beta) T - c\beta & \text{if } T < 0 \end{cases} \tag{8}$$

where $\alpha, \beta \in (0, 1)$ are increment and decrement factors, $c \in [0, 1]$ is the confidence value placed by i in the message, $\lambda \in (0, 1)$ is a forgetting factor, and $t \in [0, 1]$ is the time closeness between the current interaction and the previous one. Our calculation of experience-based trust is scalable. It updates a peer's trustworthiness in a recursive manner. The computation of our experience-based trust is thus linear with respect to the number of times receiving trust opinions from a peer. And only the most recent trust value is needed to be stored and used for computation.

The values of α and β should be subjective to road situations and message types. For example, when traffic is sparse, these values should be set larger, considering the number of trust opinions is small. For emergency related events, the values should be larger so as to increase or decrease peer trust more rapidly. Besides, it is appreciated that $\beta > \alpha$ based on the common assumption that peer trust is difficult to build up but easy to tear down.

We add the confidence c as an factor because peers, including the sender, play different roles in the message's

trustworthiness by placing different confidence values. This can be explained by the design of Equation 4, which computes the message's aggregated trustworthiness from a peer's trust and confidence. For example, between two peers with the same peer-to-peer trust, the one who has placed a confidence $c = 1$ is making greater impact than the other with a confidence $c = 0.1$. Consequently, those with higher confidence would increase or decrease their trust faster than those with lower confidence. In other words, if a peer provides a correct trust opinion, it should be praised by how much confidence it has placed in the message. The higher confidence value the peer gives, the more she should be praised. This also applies to the other direction, i.e. the punishment towards a peer who gives a wrong trust opinion.

We also model the time closeness t as

$$t = \begin{cases} (t_c - t_e)/t_{max} & \text{if } t_c - t_e < t_{max}, \\ 1 & \text{otherwise.} \end{cases} \quad (9)$$

where t_c is the current time and t_e is the event time in the message; t_{max} is the maximum time for a peer to totally forget the experience that happened before time $t_c - t_{max}$. The value of t_{max} is dependent on the frequency of the interactions between two peers in the network, and thus it should be set large under sparse traffic scenarios or small under dense traffic situations.

V. Evaluation

In this section, we present evaluation results of our trust-based framework through simulations of real life traffic scenarios by employing real maps with vehicle entities following traffic rules and road limits. We use a map of the East York area of Toronto where a snapshot of its small subarea is shown in Figure 3. Roads are partitioned into multiple road segments, and vehicles are clustered geographically by road segments. We set the length of road segment to 0.5 kilometers, because peers within such a distance can reliably communicate with each other, according to [11]. Vehicles are moving in the map in any possible directions and in different speeds. Entering a new road segment indicates that the peer is switching from one cluster to another.

Fig. 3. Map for Simulating VANET

In our experiments, the sender weight factor γ is set to 2 to double the weight of a sender in message evaluation. Assuming that peer dishonesty is well tolerated by the system, we set the peer's trust threshold τ to 0.1, and the maximum error rate ε in the relay control model to 0.8. We also set $\beta/\alpha = 10$. We simulate a total number of 1125 vehicle entities. We set 2% of them as authority roles which are fully reliable and trustworthy, and capable of providing other peers with valid observations and trust opinions.

A. Scalability

Our trust model can improve network scalability by the relay control model, which detects and filters malicious messages during propagation. We evaluate the scalability by introducing the following attack model. Attackers abuse their local vehicular network by frequently sending spam messages, which could be out-of-date information or repeated messages. Spam messages might not be misleading but they take up a certain portion of wireless resources and lower the utilization rate of available bandwidth. Our evaluation of scalability features in three metrics: average propagation distance of spam, average number of received messages per peer, and global relay effectiveness. Each evaluation metric compares the performance among six predefined scenarios as follows:

- Original: without regard to the trustworthiness of messages, they are simply relayed to the next hop, until the furthest allowed distance is reached;
- Relay Control (RC): a relay decision is made based on Equations 1 and 2 but without considering the role-based and experience-based trust;
- RC+Role: only role-based trust is involved for relay control;
- RC+Exp: only experience-based trust is used for relay control;
- RC+Role+Exp: both role-based trust and experience-based trust are used;
- 100% Detection: the ideal case where each peer detects all spam messages.

Based on the fact that the number of messages that can be relayed in a fixed period of time has an upper bound due to limited wireless channel resources, our system becomes more scalable as more normal messages can be relayed, which is achieved by detecting and controlling spam within a shorter distance. The maximum propagation distance without relay control is 5.5 km as defined in our experiment. The relay control reduces the distance of spam by nearly half, as observed in Figure 4. Authority roles further restrict the spam within approximately 2 kilometers away from origin, due to the fact that authority roles have assisted its cluster relayer to drop the spam at an earlier phase of propagation. From the curves of RC+Exp and RC+Role+Exp, we can conclude that the experience-based trust plays a greater part in spam control as our experiment simulates for a longer time. This also explains why RC+Role achieves better performance at the beginning but is sooner overwhelmed by RC+Exp after 30 minutes of system time. The curves of RC+Exp and RC+Role+Exp demonstrate the trend of converging to the performance of 100% detection, under which scenario spam is always dropped and never relayed to neighbor clusters, in other words, restricted within 0.5 kilometers (the length of

cluster defined in our experiment). As the experience-based trust of spammers is gradually decreased, their messages will not be trusted and not be relayed.

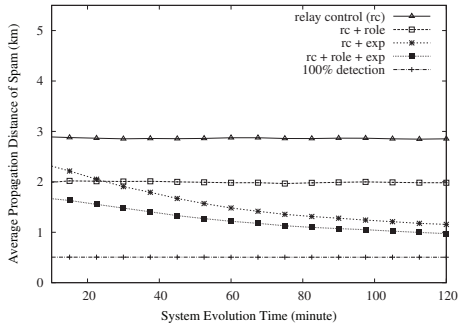

Fig. 4. Propagation Distance of Spam

We also measure the number of received messages by adjusting the ratio of spam from 0% to 100%. We track a total number of $14,400$ messages during a simulation for 2 hours. Experimental results are displayed in Figure 5. The average number of received messages decreases as the percentage of spam increases, due to the relay control model. We notice that the RC+Exp curve outperforms the RC+Role curve when the percentage of spam is greater than 23%. This is because peers learn better about spammers during a fixed period of time, as more spam messages are available when the spam ratio is raised.

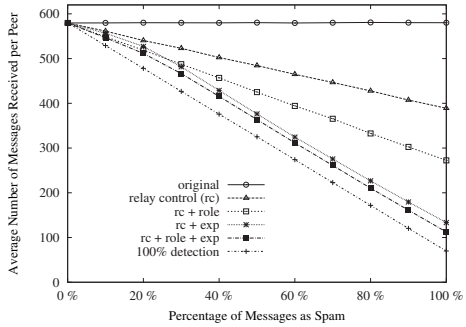

Fig. 5. Number of Messages Received

We further evaluate system scalability using the global relay effectiveness, which measures how effectively messages are relayed in the presence of a considerable amount of spam messages. Specifically, we define the global relay effectiveness $R = \frac{1}{N}\sum_{i=1}^{N} R_i$, where N is the total number of clusters, and R_i is the relay effectiveness for a single cluster C_i, which is computed as $R_i = (1 - S_i/M_i) \times 100\%$, where S_i is the number of relayed spam messages and M_i is the number of all relayed messages by cluster C_i. We illustrate the global relay effectiveness in Figure 6. Attack is suspended until 5 minutes later. Since then, as shown in the original case, the effectiveness drops to around 42% after 120 minutes. Spams are restricted from dissemination after we apply the relay control model. Role-based trust always improves the effectiveness in that spam messages are further restricted. The global relay effectiveness stops ceasing and begins to recover after 35 minutes if the experience-based trust is applied, as can be observed from curves RC+Role+Exp and RC+Exp. As peers become more experienced, the capability of the system to cope with spammers is strengthened.

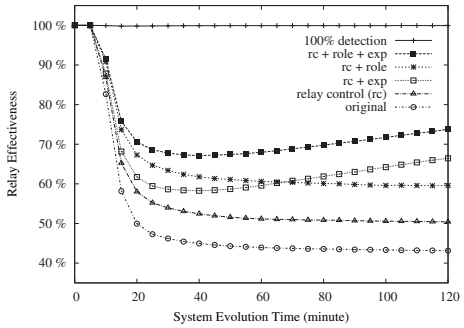

Fig. 6. Global Relay Effectiveness

B. Effectiveness

We evaluate the effectiveness of our system in terms of its capability of mitigating against malicious messages and protecting peers from being affected. We define the attack model where attackers jeopardize the network by broadcasting misleading messages on fake events, such as "traffic congestion here", so as to cheat peers and maximize their own interest. We measure the average number of wrong actions per peer. An instance of "wrong action" indicates that one malicious message is trusted by a certain peer whose action module computes an action decision of "follow" instead of "¬follow".

We measure the effect of trust opinions under three trust opinion modes:

- No trust opinions: The action module ignores all trust opinions. Specifically, when the peer is within the maximum distance where a trust opinion is available, the action module follows the reaction of the analysis module; otherwise, it simply follows the message;
- Trust opinions + majority voting: The action module computes a local action using Equation 3 without considering the trustworthiness of peers;
- Trust opinions + experience-based trust: A local action is computed from trust opinions by considering each peer's trustworthiness using Equation 4.

We run the simulation for 60 minutes and sample the data after every 5 minutes. As shown in Figure 7, each peer makes an average number of approximately 46 wrong actions if trust opinions are excluded. However, this number drastically drops to 19 (i.e. by 65%) if trust opinions are considered. The employment of experience-based trust further decreases the number of wrong actions globally as the system evolves. This is because once a peer obtains its own experience after being cheated by a malicious message, it will update the experience-based trust for those who have provided trust opinions for that message. The malicious peers' trust is shortly decreased. As a

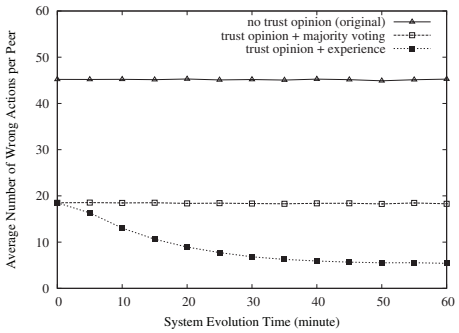

Fig. 7. Effect of Trust Opinions

result, the action module improves its accuracy by mitigating against malicious peers.

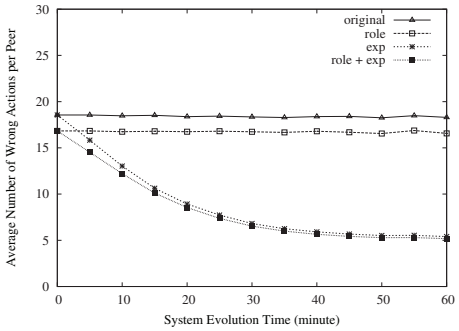

Fig. 8. Effect of P2P Trust when Relay Control Model is off

We also evaluate the effect of our peer-to-peer trust model. In our system, the peer-to-peer trust is used in both the action module and relay control model. In order to demonstrate the effect of peer-to-peer trust on the action module, we evaluate the system effectiveness under two scenarios, namely without and with the relay control model, as shown in Figures 8 and 9. In the absence of the relay control model, both good and bad messages are relayed to the furthest distance without being dropped.

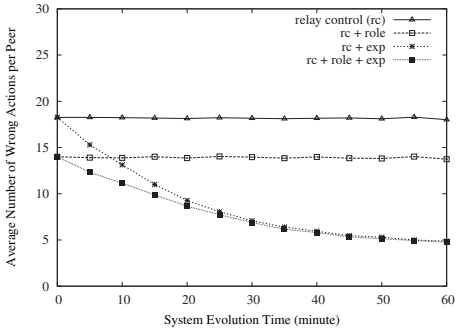

Fig. 9. Effect of P2P Trust when Relay Control Model is on

Two conclusions can be drawn from the two figures. Role-based trust improves the system effectiveness in both scenarios, due to the fact that authority roles are helpful in two ways. First, the trust opinions from authorities are always followed by the action module of peers. Since authority is always trustworthy, the number of wrong actions is decreased. Second, the trust opinions from authorities determine whether a message is to be relayed or dropped. When the relay control model is turned on, the propagation of malicious messages is limited and thus the negative effect is restricted. This explains why role-based trust decreases the number of wrong actions more in the scenario with relay control than the one without relay control. Experience-based trust improves the system effectiveness as well. As explained earlier, peers accumulate experience and lower the experience-based trust for malicious peers. As a result, the average number of wrong actions is gradually decreased as system evolves. The performance of the both curves (Exp and Role+Exp) is about the same after 60 minutes, which indicates that the experience-based trust plays a greater part in lowering the wrong decision rate than the role-based trust, as system evolves for a longer time. These results suggest that the role-based trust is especially useful when peers do not have much experience with other peers due to the data sparsity in the VANET environment or because they are new to the system. Experience-based trust is also important because it improves system performance when peers gain more experience in the environment.

VI. Related Work

The work in [12], [13] has been focused on the eviction of malicious peers via certification revocation where malicious peers will be identified and restricted from further hampering the network by the central authority. The mitigation against maliciousness is entity-oriented. In their models, the authors assume that the quality of data depends only on the honesty of the sender without considering opinions of other peers about the data. The methodology taken towards the malicious data control is reactive. Specifically, it takes a considerable time for the central authority to distribute an up-to-date revocation list before malicious peers can be timely identified. Our approach proactively detects malicious data so that the data can be immediately controlled to minimize its further negative effect on other peers.

Golle et al. [14] propose an approach to detect and correct malicious data in vehicular networks. They assume that each vehicular peer is maintaining a model which consists of all the knowledge that the peer has about the network. Data is trusted if it agrees with the model with a high probability. Otherwise, a heuristic is invoked to restore data consistency by finding the simplest explanation possible. Multiple explanations are ranked and the peers accept the data if it is consistent with the most highly ranked one(s). However, they assume that each vehicle has the global knowledge of the network and solely evaluates the validity of data, which may not be feasible in practice. Our work also provides high resistance and security against malicious entities using a fundamentally different way of message evaluation. Instead of relying on an assumed model and seeking explanations, messages in our model are evaluated in a distributed and collaborative fashion by collecting multiple opinions during their propagation.

Raya et al. [15] in their work employ trust into data evaluation in vehicular networks. In contrast to traditional views of entity-oriented trust, they proposed data-centric trust establishment that deals with the evaluation of trustworthiness of messages from other peers instead of vehicle entities themselves. A set of trust metrics are defined to represent the data trust from multiple dimensions, such as a vehicle's security status, peer type and event type. Based on Bayesian interference and Dempster-Shafer Theory, they evaluate the decision logic which outputs the trust values of various data regarding a particular event. Their work shares some commonalities with ours, such as the employment of data trust. One of the shortcomings of their work is that trust relationship in entities can never be reliably established. The data-centric trust has to be established again and again for each event, which may not be applicable to situations under the sparse environment where only limited evidence about the event is available. Our framework employs role-based trust to cope with the data sparsity problem. We also incorporate both data trust and peer trust together in our framework to detect malicious data as well as possibly malicious peers. Possibly the closest to our model, Dotzer [16] suggests building a distributed reputation model that exploits a notion called opinion piggybacking where each forwarding peer (of the message regarding an event) appends its own opinion about the trustworthiness of the data. They provide an algorithm that allows a peer to generate an opinion about the data based on aggregated opinions appended to the message and various other trust metrics including direct trust, indirect trust, sender based reputation level and Geo-Situation oriented reputation level. In our framework, we also introduce the trust-based message propagation to control the spread of malicious messages, in order to increase network scalability.

VII. Conclusion and Future Work

We presented a novel trust-based message evaluation and propagation framework in VANETs, where a set of trust metrics, including trust opinions, experience-based trust and role-based trust, are used to model the quality of information shared by peers and the trust relationships between peers. Our proposed message evaluation approach is conducted in a distributed and collaborative fashion during message propagation, and effectively increases the overall data reliability and system effectiveness by proactively detecting malicious data. We propose that message relay controls should be trust-based, filtering malicious data to promote network scalability. Experimental results demonstrate that our approach works effectively and efficiently for the domain of vehicular networks.

Our framework depends on the existence of trust opinions generated by the analysis module. The design of such a module would involve much consideration from the perspective of hardware design, such as the design of tamper-proof devices, car sensors and human-computer interactive interfaces. Our trust aggregation and message propagation model is built on a cluster-based routing scheme where cluster leaders are responsible for judging whether to relay data based on the relay control model. For future work, we will consider the presence of malicious leaders who intentionally drop messages. We will investigate a set of detection and revocation mechanisms to cope with this issue by dynamically selecting trustworthy leaders or introducing backup leaders.

In real life scenarios, it is very likely that only a subset of trust opinions are available for aggregation due to complex road settings. We will evaluate the effectiveness of our system in these cases. More complex scenarios may also be employed. For example, we will simulate the scenario where vehicle density varies to examine the capability of our system in coping with data sparsity. More sophisticated attack models may also be simulated to evaluate the resistance of our system to, for example, peer collusion attacks.

References

[1] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode, "Trafficview: Traffic data dissemination using car-to-car communication," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 8, no. 3, pp. 6–19, 2004.

[2] Q. Xu, T. Mak, J. Ko, and R. Sengupta, "Vehicle-to-vehicle safety messaging in DSRC," in *Proceedings of the ACM international workshop on Vehicular ad hoc networks*, 2004, pp. 19–28.

[3] T. ElBatt, S. K. Goel, G. H. amd Hariharan Krishnan, and J. Parikh, "Cooperative collision warning using dedicated short range wireless communications," in *Proceedings of the ACM international workshop on Vehicular ad hoc networks*, 2006, pp. 1–9.

[4] S. U. Rahman and U. Hengartner, "Secure crash reporting in vehicular ad hoc networks," in *Proceedings of the International Conference on Security and Privacy in Communication Networks*, 2007, pp. 443–452.

[5] U. F. Minhas, J. Zhang, T. Tran, and R. Cohen, "Towards expanded trust management for agents in vehicular ad-hoc networks," *International Journal of Computational Intelligence (IJCI), accepted*, 2009.

[6] T. D. Little and A. Agarwal, "An information propagation scheme for VANETs," in *Proceedings of the IEEE Conference on Intelligent Transportation Systems*, 2005.

[7] C. Gentry and Z. Ramzan, "Identity-based aggregate signatures," in *Proceedings of the International Conference on Theory and Practice of Public-Key Cryptography*, 2006, pp. 257–273.

[8] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," in *Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security*, 2001, pp. 514–532.

[9] F. Li and Y. Wang, "Routing in vehicular ad hoc networks: A survey," *IEEE Vehicular Technology Magazine*, vol. 2, no. 2, pp. 12–22, 2007.

[10] T. Tran, "Reputation-oriented reinforcement learning strategies for economically-motivated agents in electronic market environments," Ph.D. dissertation, University of Waterloo, Canada, 2004.

[11] J. Zhu and S. Roy, "MAC for dedicated short range communications in intelligent transport system," *IEEE Communications Magazine*, vol. 41, no. 12, pp. 60–67, 2003.

[12] J. J. Haas, Y.-C. Hu, and K. P. Laberteaux, "Design and analysis of a lightweight certificate revocation mechanism for VANET," in *Proceedings of the sixth ACM international workshop on VehiculAr InterNETworking*, 2009, pp. 89–98.

[13] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, and J.-P. Hubaux, "Eviction of misbehaving and faulty nodes in vehicular networks," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 8, pp. 1557–1568, 2007.

[14] P. Golle, D. Greene, and J. Staddon, "Detecting and correcting malicious data in VANETs," in *Proceedings of the ACM international workshop on Vehicular ad hoc networks*, 2004, pp. 29–37.

[15] M. Raya, P. Papadimitratos, V. D. Gligory, and J.-P. Hubaux, "On data-centric trust establishment in ephemeral ad hoc networks," in *Proceedings of the IEEE Conference on Computer Communications*, 2008, pp. 1238–1246.

[16] F. Dotzer, "Vars: A vehicle ad-hoc network reputation system," in *Proceedings of the IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks*, 2005.