SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks 2013; **6**:1–14
Published online 15 February 2012 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.519

RESEARCH ARTICLE

Trust modeling for message relay control and local action decision making in VANETs

Jie Zhang[1]*, Chen Chen[2] and Robin Cohen[2]

[1] School of Computer Engineering, Nanyang Technological University, Singapore, Singapore
[2] David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, Canada

ABSTRACT

In this paper, we present a trust-modeling framework for message propagation and evaluation in vehicular ad hoc networks. In the framework, 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 derive a local action decision about whether to follow the information by evaluating the information in a distributed and collaborative fashion while taking into account others' opinions. Experimental results demonstrate that our proposed trust-modeling framework promotes network scalability and system effectiveness, which are the two essentially important factors for the popularization of vehicular ad hoc networks, in information propagation and evaluation under the pervasive presence of false information. In particular, we clarify how our relay control serves to decrease the number of inappropriate actions taken on the basis of malicious information and enables honest peers to produce a greater number of deliveries within the network. Copyright © 2012 John Wiley & Sons, Ltd.

KEYWORDS

trust management; vehicular ad hoc networks; road safety and congestion; message relay control; local action decision making

***Correspondence**

Jie Zhang, School of Computer Engineering, Nanyang Technological University, Singapore, Singapore.
E-mail: zhangj@ntu.edu.sg

1. INTRODUCTION

With the advance and wide deployment of wireless communication technologies, vehicle manufactures and research academia have been 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, to cope with reports from malicious peers that 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 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 local action decision for the peer.

We evaluate our trust-modeling framework in simulations of real-life traffic scenarios by employing real maps. Vehicle entities involved in the simulations follow traffic rules and road limits. Some entities 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.

The rest of this paper is organized as follows. First, we give an overview of the data design and major components of our framework in Section 2. We then describe the scalable and secure trust opinion aggregation and propagation in Section 3. We also present peer-to-peer trust modeling in Section 4. We conduct experimental simulations and analysis to evaluate our framework in Section 5. After that, we survey and compare it with some related works in Section 6. Finally, we highlight conclusions and point out future research directions in Section 7.

2. OVERVIEW

The basic idea of our framework is to evaluate and disseminate a message on the basis of its quality. We design our framework in such a way that messages can be evaluated in a distributed and collaborative fashion. At the same time, the dissemination distance of a particular message is largely dependent on its quality so that our framework ensures messages of good quality to be propagated to the farthest distance while malicious data, such as spams, to be controlled to a local minimum. We model the message quality by using a trust-based approach. In other words, the quality of a message is mapped to a trustworthiness value, which can be computed from a collection of distributed feedbacks from other peers in the network. Specifically, during the message propagation, the peer who receives the message can instantly provide feedback, namely, a *trust opinion* generated from an equipped *analysis module*. A set of trust opinions is appended to the message during message propagation. For those who receive the message, their *action module* may decide to trust or distrust it by computing its trustworthiness from an aggregated list of trust opinions. Apart from the trust modeling on data quality, we further model the behavior of vehicle entities by using a *peer-to-peer trust* approach. In this section, we describe the data design and main components of our system.

2.1. Data design

Three types of messages are generated in our system: sender message, trust opinion, and aggregate message. A sender peer prepares a sender message: $M = [event, confidence, time, location]$. $confidence \in [0, 1]$ provides flexibility in reporting an event—higher confidence indicates that the sender itself is more confident in the reported event. $time \in N$ is a positive integer, and $location \in N \times N$ is a geographical coordinate, both being available from an equipped global positioning system device. Trust opinion: $O = [reaction, confidence]$, where $reaction \in \{trust, \neg trust\}$ and $confidence \in [0, 1]$, is a message provided by a peer that serves as evaluation of the sender message. Evaluation is conducted by comparing the reported event with the peer's current knowledge, which may come from a number of equipped car sensors, the local database, or even human interactions. An *analysis module* in our system provides such an opinion. Aggregate message: $A = [M, O_1, \ldots, O_n]$ is the combination of a sender message and a list of trust opinions from distinct peers.

Let us consider a simple example. A vehicle V_0 discovered a car accident and broadcasted a sender message M containing the event description "car accident," sender confidence, and time and location where V_0 spotted the accident. There are another two vehicles near V_0, namely V_1 and V_2. V_1 receives the message M and provides a trust opinion with a trust reaction and 0.8 confidence, whereas V_2 distrusts the message M and provides a distrust reaction and 0.5 confidence. The trust opinion from V_1 is then $O_1 = [trust, 0.8]$, and similarly, the trust opinion from V_2 is $O_2 = [\neg trust, 0.5]$. Aggregation on M, O_1, and O_2 can be performed by any third party, and these messages are aggregated into the aggregate message $A = [M, O_1, O_2]$. Note that sender and peer IDs are not included in the messages; instead, they are included in the signed messages. To ensure a secure data dissemination environment, we require all messages to be signed. More details about an identity-based signature scheme will be given in Section 3.1.

2.2. System components

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,

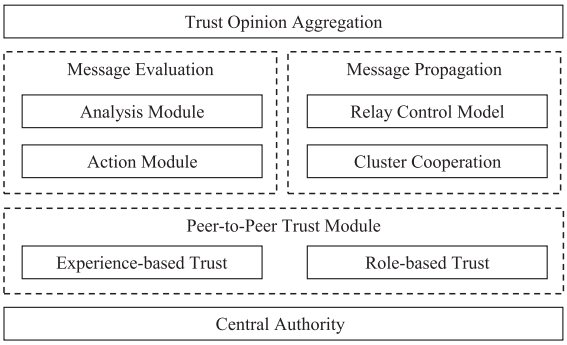

Figure 1. Design of framework.

Security Comm. Networks 2013; **6**:1–14 © 2012 John Wiley & Sons, Ltd.
DOI: 10.1002/sec

correctness, and accuracy on the basis of 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. The design of this 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. 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 by using a trust-based computation model that will be described in Section 3.3.

Message propagation consists of two components: *cluster cooperation* and *the relay control model*. On the basis of 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 performed 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.

A peer-to-peer trust module manages the trustworthiness of peers. Motivated by the approach of [5], we employ both *role-based* and *experience-based trusts*. 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.

3. TRUST OPINION AGGREGATION AND PROPAGATION

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

3.1. 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 [6–8]. By grouping peers into multiple clusters, the system becomes scalable by having message relay performed 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.

As demonstrated by an example shown in Figure 2, vehicles (peers) are geographically grouped into 10 clusters, that is, 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 preestablished 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 clusters 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 that the messages are sent by them. Aggregation should also be efficient; otherwise, it would render the system unscalable. We propose an aggregation scheme [9]

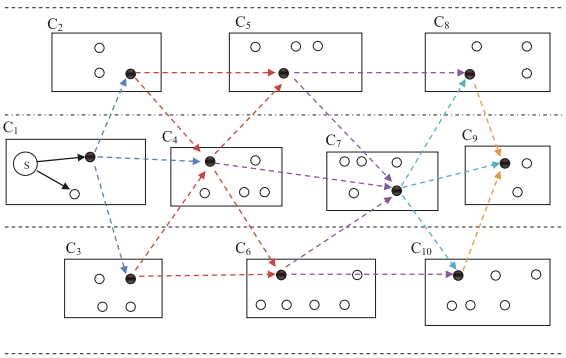

Figure 2. Cluster-based message propagation.

that extends the identity-based aggregate signature algorithm [10]. Our aggregation scheme introduces two important improvements. It can combine signatures for multiple messages (not just a single message), and it copes with signature redundancy by merging these into the existing signature that remains valid and verifiable. Thus, our proposed scheme is not only secure but also improves both space and time efficiency, with the one merged signature remaining of constant size and messages being aggregated without relying on an aggregation chain. 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 [11]. A detailed description and the verification of our identity-based aggregation scheme can be found in [9].

3.2. Message relay control

Whereas traditional routing algorithms [12] 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

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

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 4. Messages can be relayed only if

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

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; for weather information, a higher error rate can be allowed.

Algorithm 1. Message relay control

```
1: VA ⇐ a cluster leader verifies A upon reception of message A;
2: if V_A = false, then
3:   return drop;
4: else
5:   broadcasts A to cluster members;
6:   collects trust opinions Oi from cluster members;
7:   computes routing decision r using Equation 2;
8:   if r = relay, then
9:     if Δd > m_d or Δt > m_t, then
10:      //out of the maximum propagation distance or the longest
         time to live
11:      return drop;
12:    else
13:      generates A' ⇐ A + O_i + ...;
14:      return relay A';
15:    end if
16:  else
17:    return drop;
18:  end if
19: end if
```

Trustworthiness of messages ages with the time and distance. The longer time elapses and the farther the event incurs, the less accurate and reliable the data become. We use a mapping function $f_{\max} : \Lambda \times \Theta \rightarrow M_{\mathrm{t}} \times M_{\mathrm{d}}$ that maps the sender role Λ and the event Θ to the maximum time-to-live M_{t} and the 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 $10\,\mathrm{mi}^2$, whereas a life-critical message, for example "sudden brake," may only be useful within a distance of 200 m. 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: (i) verify the aggregated message A; in case verification fails, drop A; (ii) compute Δd, M_{d}, Δt, and M_{t}; if $\Delta d > M_{\mathrm{d}}$ or $\Delta t > M_{\mathrm{t}}$, drop A; (iii) compute the weight of opinion; drop A if the majority distrusts A (see Equation 2); and (iv) generate a new aggregated message A' by attaching new trust opinions of cluster members and relay A' to the next hop clusters. A pseudocode summary about how the relay control model works is shown in Algorithm 1.

Grouping peers into clusters and relaying messages between cluster leaders increase the scalability of the system considerably. 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 this important feature of our framework for vehicular networks in Section 5. An aggregated message propagated through our message propagation scheme is then used by the action module to derive an action decision for a peer.

3.3. Local action decision making

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 "¬trust." Let T_A denote the

Security Comm. Networks 2013; **6**:1–14 © 2012 John Wiley & Sons, Ltd.
DOI: 10.1002/sec

aggregated trustworthiness of the message A. The action module of peer p computes the following:

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

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 the sender having 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 computation of T_A becomes the following:

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

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. The case $\gamma = 1$ amounts to Equation 3.

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

$$T_A = \frac{\gamma c_s T_s + \sum_{i \in P} c_i T_i - \sum_{i \in P'} c_i T_i}{\gamma T_s + \sum_{i \in P} T_i + \sum_{i \in P'} T_i} \tag{5}$$

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 $f_{\text{action}} : T_A \{follow, \neg follow\}$ that maps the trustworthiness of the message to an action:

$$f_{\text{action}} = \begin{cases} follow & \text{if } T_A \geq \varphi, \\ \neg follow & \text{otherwise} \end{cases} \tag{6}$$

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 by using Equation 5, which becomes $T_A = \frac{\gamma c_s T_s + 0 + 0}{\gamma T_s + 0 + 0} = c_s$, the trust of the sender is eliminated and thus not considered. Therefore, along with the previous requirement in Equation 6 that $T_A = c_s \geq \varphi$, we further require that a peer follow the message only if $T_s \geq \tau$. A pseudocode summary about how our action module works is shown in Algorithm 2.

Algorithm 2. Local action decision making

```
1: V_A ← a peer verifies A;
2: if V_A = false, then
3:   return ¬follow;
4: else
5:   computes the value of T_A by using Equation 5;
6:   if T_A < φ, then
7:     return ¬follow;
8:   else
9:     if P = ∅ and P' = ∅, then
10://no trust opinion was provided;
11:       if T_s < τ, then
12:         return ¬follow;
13:       end if
14:     end if
15:     return follow;
16:   end if
17: end if
```

4. 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 [0, 1] denote the peer-to-peer trust of peer i, and we have

$$T_i = \begin{cases} T_i^r & \text{if peer } i \text{ has a role,} \\ f(T_{i,p}^e) & \text{otherwise} \end{cases} \tag{7}$$

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, $f(x) = (x + 1)/2$.

4.1. 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, for example, 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 that serve as part of road infrastructure; (ii) public services, which could be ambulance, fire truck, school bus, public transits, road maintenance cars, and so on; (iii) professional cars, for example, driver-training vehicles, cars whose drivers have more than 10 years of safe driving experience.

We denote the role-based trust of peer i as T_i^r, where T^r: 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.

4.2. 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 [13], if i's trust opinion leads to a correct decision of p, peer p increases the trust of i by

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

otherwise, decreases T by

$$T \leftarrow \begin{cases} \lambda^t(1+c\beta)T - c\beta \text{ if } T \geq 0 \\ \lambda^{-t}(1-c\beta)T - c\beta \text{ if } T < 0 \end{cases} \tag{9}$$

where α, $\beta \in (0, 1)$ are increment and decrement factors, c [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 $\beta > \alpha$ on the basis of the common assumption that peer trust is difficult to build up but easy to tear down.

We add the confidence c as a 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 5, 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, that is, the punishment towards a peer who gives a wrong trust opinion.

We also model the time closeness t as

$$t = \begin{cases} (t_c - t_e)/t_{\max} & \text{if } t_c - t_e < t_{\max}, \\ 1 & \text{otherwise} \end{cases} \tag{10}$$

where t_c is the current time, t_e is the event time in the message, and $t_{\max}$ 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.

5. EVALUATION

In this section, we present evaluation results of our trust-based framework through simulations. Implemented in C ++, our simulation tool allows us to simulate real-life traffic scenarios by employing real maps with vehicle entities following traffic rules, road limits, and a full list of customizable parameters defined in our trust model. We also simulate a clustering-based routing protocol in our simulation. Compared with other existing vehicular network simulation tools [14–17], our tool is specially designed for trust modeling and cluster-based messaging among potentially thousands of nodes and thus achieves more flexibility and consumes a low amount of computational resources.

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 kilometer, because peers within such a distance can reliably communicate with each other, according to [18]. Vehicles are moving in the map in

Figure 3. Map for simulating VANET.

any possible directions and in different speeds. Entering a new road segment indicates that the peer is switching from one cluster to another. A leader of a cluster is selected when the leader moves out of the cluster.

We list parameters for our trust modeling in Table I. The purposes and details of these parameters have been introduced in Sections 3 and 4. 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 $\beta/\alpha = 10$.

Additional parameters for simulating the vehicular network are listed in Table II. We simulate a total number of 1125 vehicle entities. We also set 2% of them as authority roles, such as police cars, road-side units, and traffic controllers. The authority entities are fully reliable and trustworthy and capable of providing other peers with valid observations and trust opinions. We also simulate malicious vehicle entities that always send spam messages. Vehicles in our simulation also have different capability in detecting spam messages. In consequence, they may sometimes provide wrong trust opinions.

The average number of vehicles per cluster is set to 5 to reflect the road situation during normal hours. The evaluation of the effect of traffic density is left for future work. Vehicle speed is dependent on weather conditions, traffic density, and the speed limit of the road. To simplify our experiment, we assign a unique average speed to each road, where the vehicle's speed randomly varies $\pm 10\%$ from the average speed. As mentioned in Section 2, the trust opinion is purely based on the peer's local knowledge. In our experiment, we assume that all messages are observational. From this assumption, we further assume that the analysis module can provide trust opinions only when Δd, the geographical distance between the event and the peer, is smaller than $d_{\max}$, the maximum distance for trust opinions. As a result, we can have the confidence value in the trust opinion determined by the geographical closeness: the closer the event is, the higher confidence value should be provided. In our experiment, confidence c is calculated as follows:

$$c = \begin{cases} (d_{\max} - \Delta d)/\ d_{\max} & \text{if } \Delta d < d_{\max} \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

Table I. Parameters for trust modeling.

Parameter	Description	Value
γ	Sender weight factor	2
τ	Trust threshold	0.10
φ	Action threshold	0.20
α	Experience-trust increment factor	0.01
β	Experience-trust decrement factor	0.10
λ	Experience forgetting factor	0.95
$t_{\max}$	Maximum time for experience (s)	100
ε	Error rate allowed for message relay	0.80
m_{d}	Maximum propagation distance (km)	5.50
m_{t}	Message's longest time to live (s)	150
T_0	Initial trust value of a vehicle	0

Table II. Parameters for vehicular network simulation.

Parameter description	Value
Percentage of authority roles	2%
Average number of vehicles per cluster	5
Probability of turning left/right at cross	0.2
Road segment length for one cluster	0.5 km
Maximum distance for trust opinion	1 km
Vehicle speed	[15, 30] m/s

5.1. 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. Extra parameters for the evaluation of scalability are listed in Table III. Assuming that spam is easier to detect than misleading messages as the pattern of spams has less variety, we increase the detection rate of analysis module globally by setting it to the uniform distribution from 0.4 to 1.0. We also include fewer attackers by setting the percentage of spammers to 1%, each of whom sends one spam every 5 s, which is much more frequent than misleading messages.

Our evaluation of scalability features four metrics: average propagation distance of spam, average number of received messages per peer, cumulative number of spams received 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 farthest allowed distance is reached.
- Relay control (RC): a relay decision is made on the basis of Equations 1 and 2 but without considering the role-based and experience-based trusts.
- RC + Role: only role-based trust is involved for relay control.

Table III. Extra parameters for evaluation of scalability.

Parameter description	Value
Percentage of spammers	1%
Spam sending frequency	1 message/5 s
Detection rate of analysis module	Uniform, [0.4, 1.0]

- 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.

On the basis of the fact that the number of messages that can be relayed in a fixed time has an upper bound because of 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 km away from origin because authority roles have assisted its cluster relayer (leader) 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 min 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 km (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.

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 h. Experimental results are displayed in Figure 5. The average number of received messages decreases as the percentage of spam increases because of 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 time as more spam messages are available when the spam ratio is raised.

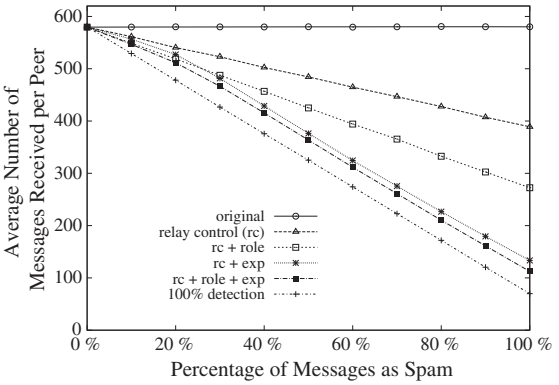

Figure 5. Number of messages received.

We then evaluate the cumulative number of spams received per peer as the system evolves. Simulation is conducted for a short duration of 50 min, as well as for a long duration of 230 min. From the simulation of a short time (see Figure 6), we can see that the RC + Exp curve is higher than the RC + Role curve until approximately 33 min later. The explanation for this is that the experience-based trust plays a greater part than role-based trust when enough experience is obtained. After simulating for a longer time (see Figure 7), the RC + Exp and RC + Role + Exp curves grow almost as slowly as the 100% detection curve, which indicates that attackers are well identified with their spam detected and controlled.

We further evaluate system scalability by 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

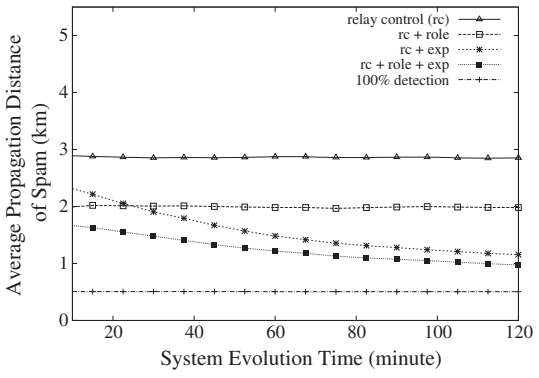

Figure 4. Propagation distance of spam.

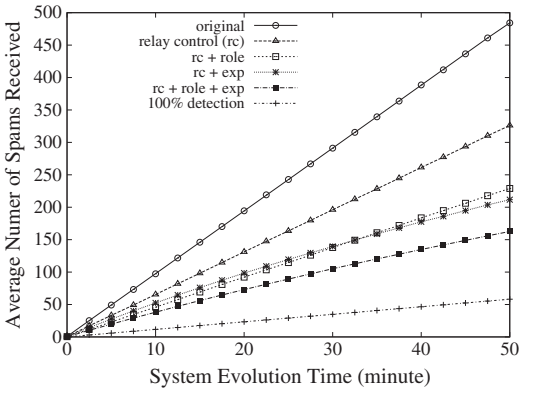

Figure 6. Average number of spams received per peer (short time).

DOI: 10.1002/sec

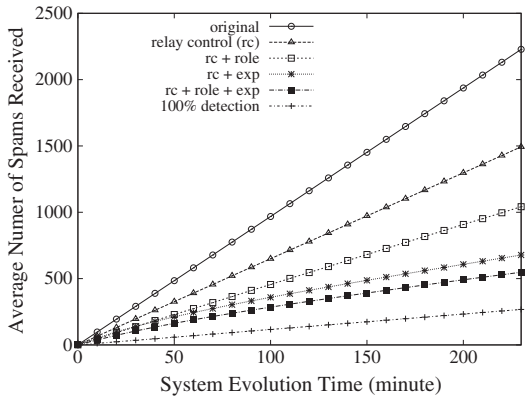

Figure 7. Average. number of spams received per peer (long time).

the number of all relayed messages by cluster C_i. We illustrate the global relay effectiveness in Figure 8. Attack is suspended until 5 min later. From then on, as shown in the original case, the effectiveness drops to around 42% after 120 min. 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 min 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.

5.2. 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".

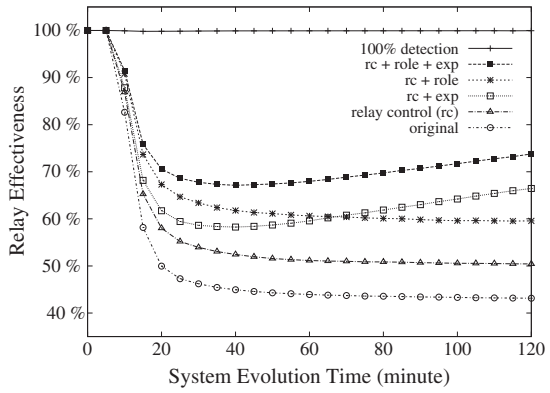

Figure 8. Global relay effectiveness.

Extra parameters for evaluating system effectiveness are listed in Table IV. Ten percent of the peers in the system are attackers, each of whom sends a malicious message after every 30 s, which is approximately the time of driving from one cluster to another. Considering that the analysis module generates trust opinions, we define the detection rate d_r as follows:

$$d_r = Pr\ \{D|M\} \tag{12}$$

where D is a successful detection given a malicious message M. The analysis module generates a trust opinion of "distrust" upon a successful detection and an opinion of "trust" otherwise. To better reflect real situations, we assume that the capability to detect malicious messages varies among peers. In our experiment, the peer's detection rate follows the uniform distribution of [0.05, 0.95], except for those in authority roles, whose detection rate is the highest and fixed to 1.

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 by 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 by using Equation 5.

We run the simulation for 60 min and sample the data after every 5 min. As shown in Figure 9, 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 result,

Table IV. Extra parameters for evaluation of system effectiveness.

Parameter description	Value
Percentage of malicious peers	10%
Frequency of malicious messages	1 message/30 s
Analysis module's detection rate	Uniform, [0.05, 0.95]

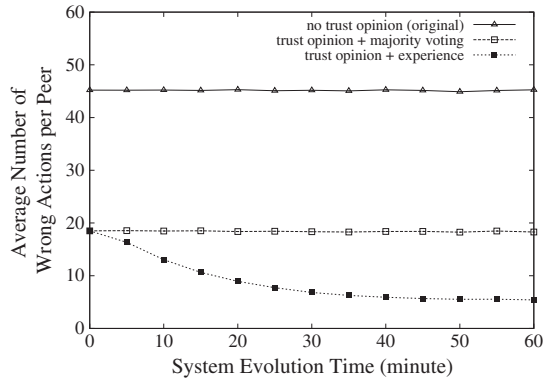

Figure 9. Effect of trust opinions.

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

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. 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 10 and 11. In the absence of the relay control model, both good and bad messages are relayed to the farthest distance without being dropped.

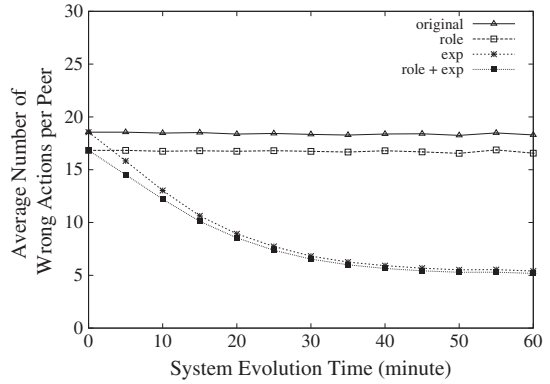

Figure 10. Effect of P2P trust when relay control model is off.

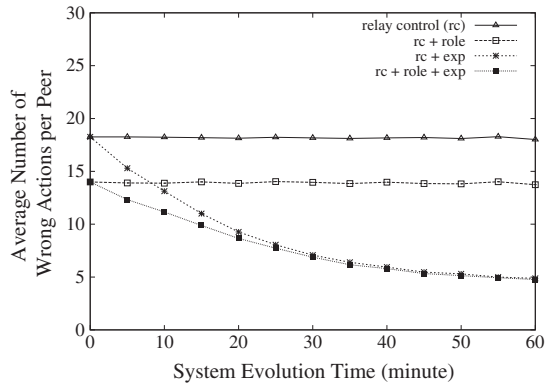

Figure 11. Effect of P2P trust when relay control model is on.

Two conclusions can be drawn from the two figures. Role-based trust improves the system effectiveness in both scenarios because authority roles are helpful in two ways. First, the trust opinions from authorities are always followed by the action module of peers. Because 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 min, 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 because of 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.

Instead of using the average number of wrong actions per peer, we use another evaluation metric "number of deliveries" to demonstrate the system effectiveness from the perspective of social impact. One delivery of the sender is defined as one message reception by some receiver. We study the social impact of peers with different honesty levels. The honesty h of a peer can be defined in possibly many ways, such as the following:

$$h = 1 - \frac{\text{number of malicious messages sent}}{\text{number of messages sent}} \tag{13}$$

We set three honesty levels in our experiment, namely 100%, 50%, and 0% honesty. Figure 12 shows the cumulative number of deliveries as the system evolves. Three peers are randomly chosen from the system, each assigned to a different honesty level. After a simulation for 20 h, it becomes obvious that the peer of 100% honesty has the largest number of deliveries because its messages are trusted and relayed to the longest distance. The cumulative curve for 0% honesty ranks the lowest because most messages from fully dishonest peers are restricted from propagation. It grows even more slowly as the system evolves because peers become more experienced so that the relay control model becomes more accurate in filtering malicious messages. Figure 13 is an alternative graph

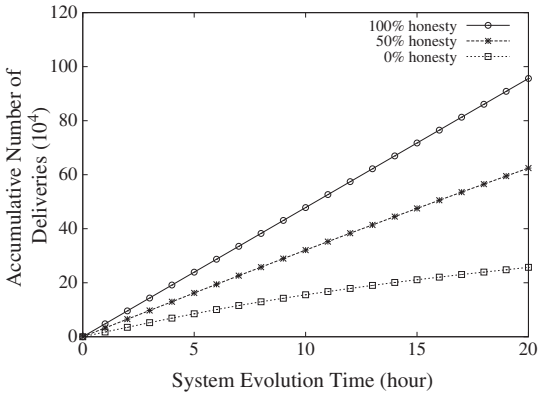

Figure 12. Cumulative number of deliveries.

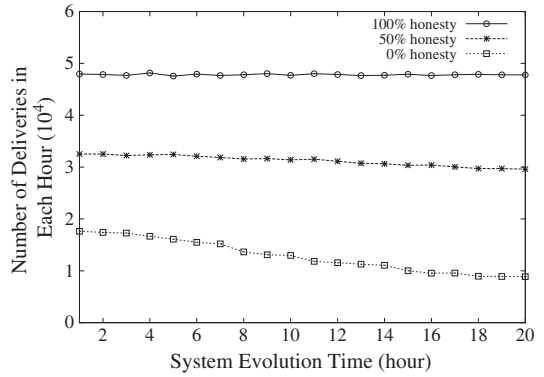

Figure 13. Number of deliveries in each hour.

showing the social impact versus peer honesty. We sample the number of deliveries for each hour and show the trend of each curve as the system evolves for 20 h. Similar to the observations in Figure 12, dishonest peers would have less social impact than honest peers.

5.3. Discussions on results

We demonstrate the system effectiveness and scalability through an experimental simulation. Throughout the experiment, we emphasize that the system effectiveness and scalability are dependent on two important factors: (i) a peer's experience with other peers; and (ii) control of malicious messages. With a better understanding and stronger control of malicious messages, system effectiveness is improved as fewer peers are affected. At the same time, the system becomes more scalable as malicious messages are more likely to be detected and dropped.

As the peer accumulates more experience and derives local decisions from a more reliable set of trust opinions, a better local decision can be made. Although the accumulation of experience can be time consuming, the experience-based trust demonstrates a strong effect on system effectiveness and scalability. The existence of authority and role-based trust improves the quality of trust opinions so that fewer wrong action decisions will be derived. The control of malicious messages is implemented by our relay control model. Although it only improves system effectiveness slightly because of the dominating effect of trust opinions, the relay control model greatly improves system scalability as it detects and filters malicious messages.

6. RELATED WORK

Vehicular ad hoc network is one of the most important applications of mobile ad hoc networks (MANETs) [19]. In this section, we first survey some trust models proposed for MANETs and point out their problems when being directly applied to the VANET domain. We then introduce a few existing trust models for VANETs and discuss the advantages of our work compared with them.

Many trust models have been proposed for MANETs [19]. For example, [20] identified several important properties of trust establishment, such as the specification of admissible types of evidences and the generation, distribution, and evaluation of trust evidences in MANETs.

A trust establishment scheme called Hermes is introduced in [21] with the objective of reliable delivery and routing in MANETs. The trust between two neighboring peers is modeled by taking into account confidence information and using a Bayesian approach based on an empirical set of first-hand observations of packet-forwarding behavior of neighboring peers. Choosing the best route between the source and destination amounts to determining the shortest path, where the weight of the path is computed from a set of peer-to-peer trust between the peers within the path. The work in [22] extends [21] in that recommendation trust is introduced to model the trust between two nonneighboring peers. The trust to a remote peer is established by collecting recommendations from a set of other peers.

Similar to [21,22], the work in [23] models the trust evaluation as a path optimization problem on a direct graph where each peer is a vertex and trust between two neighboring peer is an edge. The authors introduced the semiring-based evaluation metric that features two binary operators, + and *. The former operator is used for trust computation over a path of peers, whereas the latter is to compute the optimal aggregated trust among a set of available paths. The two operators can be reloaded via different semiring algorithms so as to adapt to various conditions.

Sun *et al.* [24,25] presented an information theoretic framework to quantitatively measure and model the trust in ad hoc networks. It first defines three trust axioms: (i) concatenation propagation of trust does not increase trust; (ii) multipath propagation of trust does not reduce trust; (iii) trust based on multiple observations from a single source should not be higher than the multiple observations from multiple independent sources. An entropy-based trust model and a probability-based model are introduced in which the author showed how to compute

trust along a path as well as the overall trust among a set of paths. The trust value between two neighboring nodes is based on observations. Third, the paper discusses how to obtain, evaluate, and update trust when it comes to ad hoc routing. Briefly speaking, each node maintains its trust record about other nodes. The source node finds multiple routes to the destination node when the source node wants to establish a route to the destination node. The source node evaluates the packet-forwarding trustworthiness of each node on a route, either by its own trust record or by requesting recommendations from other nodes. After the best trustworthy route is chosen, data is transmitted. After the transmission, the source node updates the trust records on the basis of its observation of route quality. Compared with their work, our system requires that each peer maintains a list of other peers and derives their trust from messaging and posterior experience.

The methodologies mentioned earlier may not work effectively in vehicular networks because, in practice, the trust cannot be established, maintained, or retrieved unless a reliable route is available, which is difficult to establish in a highly dynamic environment such as vehicular networks. Previous trust-modeling endeavors in MANETs, such as improving routing quality and deriving reliability between arbitrary peers, may become effortless when it comes to vehicular networks because of their two basic inherent properties. First, peer connection is ephemeral as vehicle entities are moving fast with little time for interaction. Second, interaction between two entities is highly infrequent because of a peer's mobile nature and the broad real world environment. As a result, trust establishment is difficult, and even if trust can be established between two vehicle entities, it may be out of date and uncertain. Considering the uncertainty property of trust establishment in MANETs, Balakrishnan *et al*. [26] expressed the notion of ignorance during the establishment of trust relationships between mobile nodes. A subjective logic-based model is employed to denote the trust as a three-dimensional metric: belief, disbelief, and uncertainty. The uncertainty represents the ignorance between two nodes. Such a representation is useful because an existing peer may not have a record of past evidence towards a newcomer/stranger peer, in which case assigning an arbitrary trust value could bring about problems. Compared with their work, our trust model proposes a different methodology that takes two factors into consideration, namely, a set of fixed roles and the aging factor in experience-based trust. Roles decrease the uncertainty in that their trust is fixed. The aging factor reduces the trust between two entities until new interactions are available.

Only a few trust models have recently been proposed for detecting malicious peers and data in VANETs. For example, the work in [27,28] has been focused on the eviction of malicious peers in VANETs 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*. [29] proposed an approach to detect and correct malicious data in vehicular networks. They assume that each vehicular peer is maintaining a model that 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 by 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*. [30], in their work, employed 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. On the basis of Bayesian interference and Dempster–Shafer theory, they evaluated the decision logic that 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 [31] suggested 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. He provided an algorithm that allows a peer to generate an opinion about the data on that basis of aggregated opinions appended to the message and various other trust metrics

Security Comm. Networks 2013; **6**:1–14 © 2012 John Wiley & Sons, Ltd.
DOI: 10.1002/sec

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 to increase network scalability.

7. CONCLUSION AND FUTURE WORK

We presented a novel message evaluation and propagation framework based on trust modeling for message relay control and local action decision making 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 as well as 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 also proposed that message relay control should be trust based, filtering malicious data to promote network scalability. Experimental results demonstrate that our trust-modeling approach works effectively for the domain of vehicular networks.

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 on the basis of 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.

For future work, we will vary different parameters in our simulations to more comprehensively evaluate the performance of our system. For example, in real-life scenarios, it is very likely that only a subset of trust opinions is available for aggregation because of 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. We will also simulate the situation where the aggregation of messages may take a long time and examine the robustness of our system in dealing with this situation. More sophisticated attack models may also be simulated to evaluate the resistance of our system to, for example, peer collusion attacks.

A final direction for future research would be to employ richer models of trust as part of our framework. Minhas *et al*. have recently introduced two new elements to their trust model: (i) distinguishing direct and indirect reports that are shared; and (ii) employing a penalty for misleading reports to promote honesty [32]. It would be interesting to investigate how these aspects of trust modeling would influence the message propagation within the network. It would also be useful to compare the extended trust model of Minhas *et al*. with other trust models surveyed in Section 6 to choose the most effective one for our message propagation and evaluation framework.

REFERENCES

1. Nadeem T, Dashtinezhad S, Liao C, Iftode L. Traffic view: traffic data dissemination using car-to-car communication. *ACM SIGMOBILE Mobile Computing and Communications Review* 2004; **8**(3):6–19.
2. Xu Q, Mak T, Ko J, Sengupta R. Vehicle-to-vehicle safety messaging in DSRC. In Proceedings of the ACM International Workshop on Vehicular Ad Hoc Networks, 2004; 19–28.
3. ElBatt T, Goel SK, amd Hariharan Krishnan GH, Parikh J. Cooperative collision warning using dedicated short range wireless communications. In Proceedings of the ACM International Workshop on Vehicular Ad Hoc Networks, 2006; 1–9.
4. Rahman SU, Hengartner U. Secure crash reporting in vehicular ad hoc networks. In Proceedings of the International Conference on Security and Privacy in Communication Networks, 2007; 443–452.
5. Minhas UF, Zhang J, Tran T, Cohen R. A multifaceted approach to modeling agent trust for effective communication in the application of mobile ad hoc vehicular networks. *IEEE Transactions on Systems, Man, and Cybernetics–Part C: Applications and Reviews (SMCC)* 2011; **41**(3):407–420.
6. Wu J. Dominating-set-based routing in ad hoc wireless networks. In *Handbook of Wireless Networks and Mobile Computing*. John Wiley & Sons, Inc: New York, USA, 2002; 425–450.
7. Blum J, Eskandarian A, Hoffman L. Mobility management in IVC networks. In Proceedings of the IEEE Intelligent Vehicles Symposium, 2003; 150–155.
8. Little TD, Agarwal A. An information propagation scheme for VANETs. In Proceedings of the IEEE Conference on Intelligent Transportation Systems, 2005.
9. Chen C, Zhang J, Cohen R, Ho PH. Secure and efficient trust opinion aggregation for vehicular ad-hoc networks. In Proceedings of the IEEE 72nd Vehicular Technology Conference (VTC), 2010.
10. Gentry C, Ramzan Z. Identity-based aggregate signatures. In Proceedings of the International Conference on Theory and Practice of Public-Key Cryptography, 2006; 257–273.
11. Boneh D, Lynn B, Shacham H. Short signatures from the Weil pairing. In Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security, 2001; 514–532.

12. Li F, Wang Y. Routing in vehicular ad hoc networks: a survey. *IEEE Vehicular Technology Magazine* 2007; **2**(2):12–22.
13. Tran T. Reputation-oriented reinforcement learning strategies for economically-motivated agents in electronic market environments. PhD Thesis, University of Waterloo, Canada 2004.
14. Lochert C, Barthels A, Cervantes A, Mauve M, Caliskan M. Multiple simulator interlinking environment for IVC. In Proceedings of the 2nd ACM International Workshop on Vehicular Ad Hoc Networks (VANET), 2005; 87–88.
15. Choffnes DR, Bustamante FE. An integrated mobility and traffic model for vehicular wireless networks. In Proceedings of the 2nd ACM International Workshop on Vehicular Ad Hoc Networks (VANET), 2005; 69–78.
16. Mangharam R, Weller DS, Stancil DD, Rajkumar R, Parikh JS. GrooveSim: a topography-accurate simulator for geographic routing in vehicular networks. In Proceedings of the 2nd ACM International Workshop on Vehicular Ad Hoc Networks (VANET), 2005; 59–68.
17. Saha AK, Johnson DB. Modeling mobility for vehicular ad-hoc networks. In Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks (VANET), 2004; 91–92.
18. Zhu J, Roy S. MAC for dedicated short range communications in intelligent transport system. *IEEE Communications Magazine* 2003; **41**(12):60–67.
19. Zhang J. A survey on trust management for VANETs. In Proceedings of the 25th International Conference on Advanced Information Networking and Applications (AINA), 2011.
20. Eschenauer L, Gligor VD, Baras J. On trust establishment in mobile ad-hoc networks. In *Proceedings of the Security Protocols Workshop*. Springer-Verlag: Berlin, Heidelberg, 2002; 47–66.
21. Zouridaki C, Mark BL, Hejmo M, Thomas RK. A quantitative trust establishment framework for reliable data packet delivery in MANETs. In Proceedings of the 3rd ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN), 2005; 1–10.
22. Zouridaki C, Mark BL, Hejmo M, Thomas RK. Robust cooperative trust establishment for MANETs. In Proceedings of the fourth ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN), 2006; 23–34.
23. Theodorakopoulos G, Baras J. On trust models and trust evaluation metrics for ad hoc networks. *IEEE Journal on Selected Areas in Communications* 2006; **24**(2):318–328.
24. Sun Y, Yu W, Han Z, Liu K. Trust modeling and evaluation in ad hoc networks. In Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM), 2005.
25. Sun YL, Yu W, Han Z, Liu K. Information theoretic framework of trust modeling and evaluation for ad hoc networks. *IEEE Journal on Selected Areas in Communications* 2006; **24**(2):305–317.
26. Balakrishnan V, Varadharajan V, Tupakula U. Subjective logic based trust model for mobile ad hoc networks. In Proceedings of the 4th International Conference on Security and Privacy in Communication Networks (SecureComm), 2008; 1–11.
27. Haas JJ, Hu YC, Laberteaux KP. Design and analysis of a lightweight certificate revocation mechanism for VANET. In Proceedings of the sixth ACM International Workshop on VehiculAr InterNETworking, 2009; 89–98.
28. Raya M, Papadimitratos P, Aad I, Jungels D, Hubaux JP. Eviction of misbehaving and faulty nodes in vehicular networks. *IEEE Journal on Selected Areas in Communications* 2007; **25**(8):1557–1568.
29. Golle P, Greene D, Staddon J. Detecting and correcting malicious data in VANETs. In Proceedings of the ACM International Workshop on Vehicular Ad Hoc Networks, 2004; 29–37.
30. Raya M, Papadimitratos P, Gligory VD, Hubaux JP. On data-centric trust establishment in ephemeral ad hoc networks. In Proceedings of the IEEE Conference on Computer Communications, 2008; 1238–1246.
31. Dotzer F. 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.
32. Minhas UF, Zhang J, Tran T, Cohen R. Intelligent agents in mobile vehicular ad-hoc networks: leveraging trust modeling based on direct experience with incentives for honesty. In Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT), 2010.