A Multifaceted Approach to Modeling Agent Trust for Effective Communication in the Application of Mobile Ad Hoc Vehicular Networks

Umar Farooq Minhas, Jie Zhang, Thomas Tran, and Robin Cohen

***Abstract*—An increasingly large number of cars are being equipped with global positioning system and Wi-Fi devices, enabling vehicle-to-vehicle (V2V) communication with the goal of providing increased passenger and road safety. This technology actuates the need for agents that assist users by intelligently processing the received information. Some of these agents might become self-interested and try to maximize car owners' utility by sending out false information. Given the dire consequences of acting on false information in this context, there is a serious need to establish trust among agents. The main goal of this paper is then to develop a framework that models the trustworthiness of the agents of other vehicles, in order to receive the most effective information. We develop a multifaceted trust modeling approach that incorporates role-, experience-, priority-, and majority-based trust and this is able to restrict the number of reports that are received. We include an algorithm that proposes how to integrate these various dimensions of trust, along with experimentation to validate the benefit of our approach, emphasizing the importance of each of the different facets that are included. The result is an important methodology to enable effective V2V communication via intelligent agents.**

***Index Terms*—Intelligent agent, intelligent vehicles, multiagent systems.**

I. Introduction

With the advancement in technology, more and more vehicles are being equipped with global positioning system and Wi-Fi devices that enable them to communicate with each other, creating a vehicular ad hoc network (VANET). Various studies have established the fact that the number of lives lost in motor vehicle crashes worldwide every year is by far the highest among all the categories of accidental deaths [1]. It is apparent that there is a dire need to enhance passenger and road safety, which is precisely one of the goals of deploying vehicle-to-vehicle (V2V) communication systems. Another supporting goal is to be able to effectively route traffic through dense urban areas by disseminating up to date information regarding road condition through the VANET. Network-on-wheels project [2], GST, PreVent, and car-to-car consortium [3] among others, represent some of the ongoing efforts in the general domain of vehicular networks.

Some car manufacturers have already started to fit devices that will help to achieve the goals mentioned earlier. For example, General Motors (GM) has rolled out V2V communication in its Cadillac STS Sedans.GM's proprietary algorithm called "threat assessment algorithm" keeps track of the relative position, speed, and course of other cars (also equipped with V2V technology) in a quarter-mile radius and issues a warning to the driver when a crash is imminent [4]. Similar prototypes by other car manufacturers are currently in the testing phase, scheduled to hit the markets over the coming years.

Even though the initial algorithms and protocols that are being proposed by the car manufacturers are proprietary, it is believed that the standardization efforts carried out by car-to-car consortium [3] will help to define a common interface for V2V communication technologies allowing its wide-spread use. Following this, it is very natural to assume that agent applications will be deployed, whose main goal will be to assist the user in various ways using V2V communication. One such example is of an agent that gathers road congestion information and calculates the optimal route from a user's origin to destination, thus bringing utility to the user. In such a scenario, we can view cars in a VANET as autonomous agents acting on behalf of their owners, thus constituting a multiagent network.

The agent would represent the motives of car owners, who might as well decide to behave selfishly every now and then. For example, consider a user, who instructs his agent to report the roads on his path as congested with the hope that other agents would avoid using these roads, thus clearing the path. Therefore, one important issue among others that may arise in VANETs is the notion of trust among different agents. The goal of incorporating trust is to give incentives for these agents to behave honestly and to discourage self-interested behavior. These details are captured through what is called a *trust model*. Defined formally, "trust is a belief an agent has that the other party *will do what it says it will* (being honest or reliable) or *reciprocate* (being reciprocative for the common good of both), given an opportunity to defect to get higher payoffs" [5]. A closely related notion called reputation is defined as the opinion or view of an agent about another agent, i.e., either directly acquired from the environment or from other agents and ultimately leads to building of trust [5]. Given the critical nature of agent applications

Manuscript received April 17, 2009; revised November 18, 2009, April 30, 2010, and September 16, 2010; accepted September 22, 2010. Date of publication November 22, 2010; date of current version April 19, 2011. This paper was recommended by Associate Editor L. B. Sheremetov.

U. F. Minhas and R. Cohen are with the David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, ON N2L3G1, Canada (e-mail: ufminhas@cs.uwaterloo.ca; rcohen@ai.uwaterloo.ca).

J. Zhang is with the School of Computer Engineering, Nanyang Technological University, Singapore 639798 (e-mail: j44zhang@cs.uwaterloo.ca).

T. Tran is with the School of Information Technology and Engineering, University of Ottawa, Ottawa, ON K1N 6N5, Canada (e-mail: ttran@site.uottawa.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCC.2010.2084571

in the context of VANETs, it is crucial to associate trust with agents and the data that they spread.

Modeling trustworthiness of agents in VANETs presents some unique challenges. First of all, the agents in a VANET are constantly roaming around and are highly dynamic. On a typical highway, the average speed of a vehicle is about 100 km/h. At high speeds, the time to react to an imminent situation is very critical [6], therefore, it is very important for the agents to be able to verify/trust incoming information very quickly. Second, the number of agents in VANET can become very large. For example, in dense urban areas, the average amount of vehicles that pass through the network may be on the order of millions and several thousand vehicles will be expected to be present in the network at any given time. Also, this situation is exacerbated during the rush hours when, for example, majority of the people commute to and back from work in a metropolitan area. This may introduce several issues some of which include network congestion—since vehicles are communicating on a shared channel, information overload—resulting from vehicles receiving a lot of data from the nearby vehicles in a congested area, etc. Hence, there will be a need to have intelligent vehicle communication systems that are *scalable* and can detect and respond to these potentially hazardous situations by effectively deciding with which agents to communicate [7].

Another key challenge in modeling trust in a VANET environment is that a VANET is a *decentralized*, open system, i.e., there is no centralized infrastructure and agents may join and leave the network any time, respectively. If an agent is interacting with a vehicle now, it is not guaranteed to interact with the same vehicle in the future [8]. However, a social network of agents may still emerge (for example, a commuter pool of agents). This suggests that agents may best manage their own communication with other agents, as information is needed, instead of relying on a centralized system for the management of all the information.

Also, information about road condition is rapidly changing in VANET environments, e.g., a road might be busy 5 min ago, but now it is free, making it hard to detect if the agent spreading such information is malicious or not. This also brings out an important challenge that the information received from VANETs needs to be evaluated in a particular context. The two key context elements in VANETs are *location* and *time*. Information, which is closer in time and location of an event is of more relevance. We explain this in more detail in Section III.

Various trust and reputation models (e.g., [9] and [10]) have been studied with reference to multiagent environments, however, given the unique characteristics of agents in VANETs, the existing models cannot be applied directly. For example, several trust and reputation models are built around the assumption that the agents can have multiple direct interactions with other agents, and hence, they fail when applied to VANETs. Key issues with these models will be explained in a greater detail in Section II.

The main goal of this paper is then to develop a framework that can effectively model the trustworthiness of the agents of other vehicles in VANETs. We propose a novel multifaceted approach for modeling trust in VANET environments that incorporates role-, experience-, priority-, and majority-based trust and is able to restrict the number of reports that are received from other agents. Our expanded trust model is aimed to be decentralized, location/time specific, event/task specific, able to cope with the data sparsity problem, cumulative in order to be scalable, sensitive to privacy concerns, and able to support system-level security. We present the design of this model in detail, clarifying how it meets various critical challenges for trust modeling in VANET environments. We also step through a detailed procedure of computing trustworthiness of agents and generating effective responses to information sent by these agents. We finally demonstrate its value in a simulated vehicular setting. The result is an important first step toward the delivery of effective intelligent vehicular communication, one that is sensitive to the trustworthiness of the vehicular agents.

The rest of this paper is organized as follows. Section II identifies some key areas, where existing models lack in their applicability to VANETs. In Section III, we present the design and implementation of our proposed expanded trust model for VANETs. In Section IV, we describe the results of experiments carried out in a simulated vehicular setting. In Section V, we present some related work, and finally, Section VI provides the conclusion and future work.

II. Key Issues With Existing Trust Models

In this section, we identify some of the key issues with the current trust models proposed for multiagent systems that render them ineffective, completely or to a certain degree, when taken to the domain of VANETs. The discussion of these issues inspires the design and implementation of our particular framework for effectively modeling trust of agents in VANET environments.

A. Trust Emerging From Multiple Direct Interactions Between Agents

Many trust models proposed in literature have an underlying assumption that agents interact multiple times with each other over a period of time. In learning and evolutionary models of trust, such as those presented in [9]–[14], an agent learns to trust (or distrust) another agent based on its past interactions with another agent. If the past interactions with a particular agent have been particularly rewarding, the other agent would then learn to associate a higher trust value resulting in a higher chance of future interactions with this agent. However, if a certain agent is known to defect over the past interactions, the other agent will choose not to deal with it in the future, thus representing a lower (learned) value of trust. For a more concrete example, the model of Tran [14] is presented for the domain of multiagent-based e-marketplaces.They have buying agents, who use reinforcement learning to determine with which selling agents to do business, in order to maximize the buyers' expected profit. They also have selling agents, who use the same learning method to maximize the sellers' profit by adjusting product prices and altering product quality offered to different buyers. To avoid doing business with possibly dishonest sellers, buyers in the market determine the trustworthiness of the sellers using an incremental updating approach, after the true value of delivered products is evaluated and compared to the buying agent's expected value for the products. This approach updates the trustworthiness of sellers based on their previous trust values after examination of goods. The trustworthiness of sellers will be reliably learned after buyers have had multiple direct interactions with sellers.

In these models, having multiple direct interactions among agents is the key to establishing trust and in learning to evolve strategies over time. However, in highly dynamic and open multiagent systems, such as VANETs, it is not logical to expect that this assumption will hold. Therefore, while it may be valuable to incorporate an element of direct experience in VANET environments (e.g., for the case of vehicles following the same road on a regular basis, as commuters), the trust models, whose success depends on a certain minimum number of direct interactions between the agents, are not sufficient for the domain of VANETs.

As we argued earlier, effective trust establishment should not be contingent upon a minimum threshold for direct interactions. However, a trust model for VANET should still be able to effectively take into consideration any data available from direct interactions (even though it might happen just once). The evidence from direct interactions, whenever available, can be very easily incorporated into trust calculation as part of our multifaceted framework that will be described in Section III.

B. Degree of Knowledge About the Environment

Majority of the learning and evolutionary models of trust presented in the literature for multiagent systems, such as [11]–[13], [15], assume complete information about other agents and the system (e.g., strategies, payoff matrix, etc.) in order to make their trust learning algorithms work. This assumption might hold in certain restrained scenarios (such as controlled simulations), but is simply not true in VANETs, where agents are inherently limited in their capacity to gather information from other agents or the environment. Though this issue arises in any multiagent environment, where there is some degree of uncertainty about other agents and the environment, we believe that it is of far more concern in the domain of trust for VANETs and we also attribute it to the rapidly changing dynamics of the agents/environment in the context of VANETs.

Most of the trust and reputation models are proposed for the domains, such as e-marketplaces, chat rooms, online auctions, etc., where the environments are rather stable, i.e., the number of agents present remains more or less constant. These models assume a static environment or allow limited dynamism if at all [12], [16]–[20]. Certain models [9], [21], [22] have been proposed to deal with this issue to some extent. For example, the Bayesian network-based model of Regan *et al.* [9] considers a particular scenario, where buying agents trying to choose selling agents based on the opinions of other buying agents (advisors) that have had past interactions with the selling agents. They propose that the evaluation function used by the advisors in reporting the ratings of the sellers can be learned over time by the buying agent, and then, can be used to produce a personalized reinterpretation of the ratings reducing the effects of a buyer's subjectivity and deception and the change in buyer and seller behavior.

However, we believe that these models still lack in their applicability to VANETs essentially because of the rate at which agents are moving around (average 100 km/h) and joining or leaving the network is unparalleled to any other setting. Furthermore, none of these models have been shown to work (or not to work) for VANETs. We propose that any good trust model for VANETs should introduce certain dynamic trust metrics using which it can capture the changes in the environment by allowing an agent to control trust evaluation depending on the situation at hand [23], [24]. We discuss such dynamic metrics in Section III.

C. Exploiting the Social Network

Various reputation models [17], [19], [20], [25], [26] that have been presented over the past exploit in one way or another, the notion of social network, i.e., the concept of group or neighbor. A model of trust based on individual and group ratings is presented by Wang and Vassileva [19] for establishing trust among interest-based communities in peer-to-peer file sharing networks. The particular application that they explore is sharing academic research papers with other agents in the network. Agents share research papers along with their ratings regarding the quality of paper. Other agents can then search the network for papers and select based on the ratings. To exploit the social network, agents can create communities based on their interest, and then, invite other agents to join the community that they deem trustworthy and possibly have the same level of interest and knowledge. Here, the notion of trust is the ability of an agent to share high-quality research papers. An agent A trusts an agent B, if over the past agent A has liked the papers shared by agent B and found the ratings associated with the paper similar to the ratings in its own view. Similarly, this notion is extended for trust between an agent and a community. They have also provided mechanism to calculate and update trust in agents as well as aggregated trust of a community.

The success of these reputation models depends on the existence of certain connections between the agents in order to be able to reliably gather opinions from them and ultimately associate trust with the unknown agents (or to gradually build a model of the social network itself). However, given the inherently temporary nature of relationships in VANETs, it is not logical to expect that we would be able to define any meaningful relationship between different agents, thus rendering all of these reputation models ineffective to certain extend.

Though the reputation models mentioned earlier cannot be directly (and completely) applied to the agents in VANETs mainly because there are no long-term relationships (or connections) between agents, we believe that the trust models for VANETs can exploit certain predefined roles that are enabled through the identification of agents (vehicles). For example, agents can put more trust in certain agents as compared to others, e.g., in agents identified as law enforcing authorities or owned by government [24]. We discuss these roles and how they can be incorporated in our expanded trust model for VANETs in more detail in Section III.

D. Role of Central Entities

Some of the reputation models and security mechanisms depend on a central entity (or authority) to gather and aggregate opinions or to authenticate an agent based on a certificate from a central certification authority (CA). However, in a decentralized open system, such as VANETs, the assumption to have a central authority that is accessible to and trusted by all the agents will not hold. Even if for a moment, we assume that we can implement a central CA that overlooks all the agents present in the VANET, given the number of agents expected to be present in the network, the certification list will grow to the extent that authenticating an agent by consulting this central authority (i.e., searching the list of certificates) in real time would become infeasible not to mention that some models require consulting multiple authorities.

We propose that trust establishment should be fully decentralized to be applicable to the highly dynamic and distributed environment of VANETs [20], [21], [23]. If the use of certificates is desired (for referrals), that should be done in a totally decentralized manner among the vehicles themselves [16], [21]. Mass and Shehory [21] provide a model that on seeing a certificate enables a third party (or agent) to assign specific roles to the agents in the system. Based on their roles, the agents are then supposed to carry our certain duties and are expected to abide by certain policies. In this scenario, any agent can act as a certificate issuer, and thus, role assignment is achieved in a distributed fashion.

In our case, we might have to involve the car manufacturers, or transportation authorities to issue these certificates at the manufacture or registration time, respectively. Also, there would be a need to store these certificates in a way that they cannot be manipulated or tampered with [27]. The detailed implementation of this will be further discussed in Section III, along with the brief mention of privacy concerns.

III. Expanded Trust Management

In this section, we first present the design of our expanded trust management for modeling trustworthiness of agents and aggregating their feedback in VANET. We highlight some key properties that our management is aimed to have, in order to be particularly suitable for the problem domain of VANET. Then, we step through a detailed procedure of computing experience-, role-, and priority-based trust, and finally, the majority feedback and associated confidence values.

A. Design

From the discussion in previous sections, it becomes apparent that no single trust or reputation mechanism can work particularly well for the challenge of modeling trust effectively for VANET environments. Instead of just having one or two trust metrics for evaluating trust, there is a need to have several different trust metrics with various key properties in order to capture the complexity that arises between interacting agents in VANET. We propose that in order to derive a rather complete and comprehensive view of trust for agents in VANET, we will need to integrate security solutions (at the system level) for trust management, i.e., secure storage of role identities for role-based trust in our proposal.

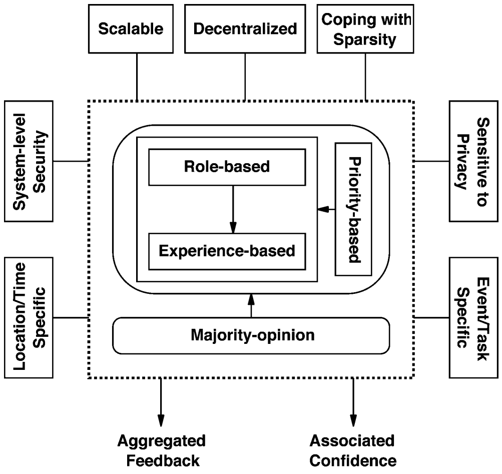


Fig. 1. Expanded trust management.

Fig. 1 illustrates the design of our expanded trust management. The core of the management is grouped by the dashed rectangle in the middle. This core consists of two parts. One part maintains trustworthiness of agents in order for trusted agents (advisors) to be chosen to ask for their feedback. More specifically, in this part, the trustworthiness of agents is modeled based on role- and experience-based trust, which are both combined into the priority-based model that can be used to choose proper advisors.

Our role-based trust exploits certain predefined roles that are enabled through the identification of agents (vehicles). For example, agents can put more trust in certain agents as compared to others, i.e., agents identified as law enforcing authorities or owned by government [24]. Our experience-based trust represents a component of trust that is based on direct interactions. It is in the same spirit of incorporating evidence from direct interactions into trust calculation through interaction trust as proposed by [16] or the individual dimension of trust in the model as proposed by [17]. Implementation and formalization of these two trust metrics will be presented in Section III-C.

The other part of the core is a majority-opinion approach to aggregate feedback from selected advisors. Detailed procedures for these processes will be further discussed in Section III-C. More importantly, our management of trust has several key properties represented by rectangles around the core in the figure. Our trust management is aimed to be decentralized, location/time specific, event/task specific, able to cope with the data sparsity problem, cumulative in order to be scalable, sensitive to privacy concerns, and able to support system-level security. These properties will be extensively discussed in Section III-B, respectively. Note that the property of system-level security is mentioned in different places, where we discuss other properties and our model, i.e., secure storage of role identities in

Section III-B1, verification of time/location of reported events in Section III-B3, and identification of agents' roles in Section III-C3.

The outcome of our trust management is aggregated feedback for a certain request/event and an associated confidence value for it. The aggregated feedback is eventually affected more heavily by highly trusted advisors. The value of confidence would depend on the reliability of estimated experience-based trust of each other agent and the maximum accept error rate for the aggregated feedback. In general, a higher value of confidence, i.e., a value closer to 1, would result from considering more evidence or metrics having high reliability, for a fixed error rate. We can view confidence as a parameter that adds another dimensionality to the output generated by the model allowing the agent applications to have a richer notion of trust, and finally, decide how to react on the reported event. Our notion of confidence is somewhat tantamount to the notion proposed in [16] and [22].

B. Key Properties

We provide here detailed discussion of the seven key properties that our trust management incorporates. These properties guide our design of the expanded trust management, which can be applied to the problem of trust management in VANET.

1) Decentralized Trust Establishment: Models, which depend on a central entity for the reliable establishment of trust are not desirable for the domain of VANET because of its highly distributed property. Therefore, we propose that trust establishment should be fully decentralized to be applicable to the highly dynamic and distributed environment of VANETs [21], [23], [28].

Our experience-based trust model makes use of agents' direct interactions to update one agent's belief in the trustworthiness of another. This one-to-one interaction can easily be implemented in a distributed manner. Our role-based trust can also be done in a totally decentralized manner among the vehicles themselves. For this to work, we may involve the car manufacturers, or transportation authorities to issue certificates at the manufacture or registration time, respectively. For example, we could use a public–private key infrastructure (PKI) for verifying each other's *roles* implemented in a distributed manner.[1] Also, there would be a need to store these certificates and keys in a way that they cannot be manipulated or tampered with, to archive high security. To this end, researchers [27], who have done studies with the goal of securing VANET communications have unanimously proposed the use of a tamper proof device that stores, e.g., the cryptographic keys issued by authorities. If any attempt to tamper the device is made, the keys are destroyed automatically stripping the agent from its ability to communicate with other agents, thus effectively destroying its means of deriving any utility at all.

2) Coping With Sparsity: Effective trust establishment should not be contingent upon a minimum threshold for direct interactions. As we have described at several places, it should not be expected that an agent in VANET would possibly interact with the same agent more than once. However, it is important to clarify here that the trust models should still be able to effectively take into consideration any data available from direct interaction (even though it might happen just once). Thus, in a scenario, where the number of agents that are able to spread information has gone down to the extent that the condition of information scarcity or a total lack of information is prevalent, any data might be termed valuable. In the trust calculation, the weight for the data can be raised in this scenario, while it may have a lower default value, to cope with the data sparsity problem in VANET.

We also have the role-based trust approach to distinguish trustworthy agents from untrustworthy ones to some extent. When an experience-based trust approach is used, inspired by the work in [29] and [30], we also introduce the idea of allowing agents to send testing requests to deal with sparsity. The senders of these testing requests basically know the solution to these requests in advance. Imaging a group of agents driving in a city from one location to another, they remain in contact range for a certain period of time. These agents can send testing requests to each other and evaluate their feedback. Trust between them can then be established through the experience-based trust in our management model.

3) Event/Task and Location/Time Specific: Since the environment of the agents in VANET is changing constantly and rapidly, a good trust model should introduce certain dynamic trust metrics, capturing this dynamism by allowing an agent to control trust management depending on the situation at hand [23], [24]. Here, we separately deal with two particularly important dynamic factors in the context of VANETs, event/task, and location/time.

Agents, in general, can report data regarding different events, e.g., car crashes, collision warnings, weather conditions, information regarding constructions, etc. Our trust management should, therefore, be event/task specific. For example, some of these tasks may be time sensitive and require quick reaction from the agent that receives them. In this case, this agent can only consult a very limited number of other agents to verify whether the reported information is true. In another case, reporting agents having different roles in VANET may have more or less knowledge in different types of tasks. For example, a police may know more about car crash information, while city authorities may know more about road construction information. Thus, our role-based trust should be task specific. One way to implement this in our role-based trust model is to have a set of events associated with a set of roles of agents (e.g., law enforcement and municipal authorities). This information can be obtained from a transportation authority and be used later for an agent to choose particular other agents to consult regarding a particular event. Our experience-based trust is also event specific. An agent updates the reporting agent's trust by taking into account the type of the reported event. For example, life-critical events will certainly have more impact on the reporting agent's trust.

We also note that location and time are another two particularly important dynamic metrics. For example, if the origin of a certain message is closer to the location of where the reported

[1] Note that PKI is important in this distributed environment, for example, when dealing with the problems, where several drivers may share a single car and the driver of a car will be changed upon selling the car.

event has taken place, it might be given a higher weight, relying on the underlying assumption that an agent closer to the event is likely to report more realistic data about the event (given that they are not malicious themselves). Similarly, we can apply this concept to time. If the message reporting a certain event is received closer to the time when the reported event has taken place (e.g., message indicating a road is free right now, compared to one reported by an agent, who observed it half an hour ago), it might be allowed a higher weight in trust calculation. Another suggestion that naturally follows from time-based trust is that, since the relevance of data in VANET is highly dependent on when it was received, it would make sense to assign a decay factor to the message. The message further away from the time of evaluating trust would be assigned a lower weight. In other words, we should decay the impact of message relative to the time of the trust evaluation. The decay factor is somewhat analogous to the time-to-live (TTL) field used in IP packets.

The first issue that may arise with calculating time- or location-specific trust is how to get location and time of the actual event. We expect that whenever a report regarding an event is generated to be shared among other agents, it will hint to the time at which this event has taken place, giving us the required time information. Also, we assume that every agent while transmitting the report, appends its location with the report. The next issue is to verify whether the time and location information contained within a report is real or spoofed. With this regard, Golle *et al.* [31] have proposed a method to accurately estimate the location of nearby agents. However, complete treatment of this issue is beyond the scope of this paper. Now the next task would be to actually use the location/time information in trust management. In the calculation of subjective reputation as proposed by [17], they use a weighted sum of trust values suggesting that the weights should be adjusted such that higher weights are assigned to the agents closer to the agent, which is calculating trust. In a similar fashion, we can extend their model by instead of defining the closeness between agents; we define the location closeness between the actual event and the agent reporting this event. For the time-based trust, a similar calculation can be done by modifying the notion of time closeness as that between the time when the event has taken place and that of receiving the report.

4) Scalable: Scalability is an important aspect in trust management in VANET environments. In our system, each agent consults only a number of other trusted agents. This number can be fixed or slightly updated with the changes in, for example, VANET size or the task at hand. However, it is always set to a value small enough to account for scalability.

Establishing trust in VANETs using experience-based trust requires each agent to store the history of past interactions with other agents and to compute their trust based on that information. For the purpose of being scalable, our experience-based trust model updates agents' trustworthiness by accumulatively aggregating agents' past interactions in a recursive manner, similar to [32]. The computation of our experience-based trust is thus linear with respect to the number of interactions, and only the most recent trust values are needed to be stored and used for computation. This design makes our trust management scalable.

5) Sensitive to Privacy Concerns: Privacy is an important concern in a VANET environment. In this environment, the revealing of a vehicle owner's identity (e.g., the owner's home address) may allow a possibly malicious party to cause damage to the owner. Our trust management could be integrated with a PKI allowing agents to authenticate each other. In our system, when an agent sends a report to another agent, the sender would need to authenticate itself to the receiver that it has a certain role. We may additionally introduce methods as in [33] to also allow for the changing of keys. How best to address privacy issues is not the primary focus of our paper and we leave this for future investigation.

C. Computation Procedure

In this section, we briefly outline the procedure taken by an agent to make a decision for a (requested) task/event by aggregating reports about this task from other trusted agents and to update their experience-based trust values afterward.

1) Scenarios: An agent in a VANET environment may passively wait for other agents to send reports about an event. In many scenarios, the agent will instead actively send a request to a list of trust neighboring agents to inquire about a task. A sample message would be "Is road X congested with traffic?," expecting a yes/no response from the other agents being asked. Once it receives a report about an event from another agent, it may trust the information if it has high confidence that the report sender can be trusted. Otherwise, it may need to verify (double check) if the information given by the sender is reliable by asking other trusted agents. In both scenarios, the agent will need to aggregate senders' reports. Values calculated in this manner can then be used by the agent to decide whether to believe a particular report and take corresponding actions. For this purpose, each agent in our system keeps track of a list of other agents.[2] This agent updates all senders' trustworthiness report after the truth of their reported events is revealed. The aforementioned two processes of aggregating reports and updating trust will take into account the context in general, this agent's notion of which other agents it is interacting with, the notion of which group the other agents belong to or the roles assigned to the other agents, the time of reported event together with the time of message arrival, the relative locations of the other agents, and the actual contents of the message to evaluate task/event-specific trust, etc. Next, we provide detailed description and formalization of each step in our computation procedure.

2) Computation Steps: Four elements are incorporated into our overall trust management as its core, shown in Fig. 1: 1) experience-based trust; 2) role-based trust; 3) majority opinion (or social network of trust); and 4) priority-based trust. Our computation procedure consists of four steps.

Step 1: Depending on the task at hand, set a value $n =$ number of agents, whose advice will be considered. This incorporates task-based trust. For example, if you need a very quick reply,

[2] The number of the other agents depends on the agent's capability and resource limit. With high-computation power and large memory size of a computer equipped in car, the capacity should be fairly large.

you may limit $n \leq 10$; if you are planning ahead and have time to process responses, n could potentially be larger.[3]

Step 2: Using n, construct an ordered list of agents to ask. The list will be partitioned into groups as follows[4]:

$$\begin{bmatrix} G_1: & a_{11}, & a_{12}, & a_{13}, & \dots, & a_{1k} \\ G_2: & a_{21}, & a_{22}, & a_{23}, & \dots, & a_{2k} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ G_j: & a_{j1}, & a_{j2}, & a_{j3}, & \dots, & a_{jk} \end{bmatrix}$$

where $jk = n$. This priority list is ordered from higher to lower roles, for example, G_1 being the highest role. Within each group of agents of similar roles, the group is ordered from higher (experience-based) ratings to lower ratings. Thus, a_{ij} represents the agent in role class i that is at the jth level of experience, relative to other agents at that level. Hence, role- and experience-based trust are combined into this priority-based approach. These two trust metrics will be further discussed later in this section.

Step 3A: When an agent requires advice, the procedure is to ask the first n agents the question, receive the responses, and then, perform some majority-based trust measurement.

Step 3B: The processing of the responses is as follows: if there is a majority consensus on the response, up to some tolerance that is set by the asker (e.g., I want at most 30% of the responders to disagree), then this response is taken as the advice and is followed. We will formalize this majority-based trust in Section III-C5.

Step 3C: Once this advice is followed, the agent evaluates whether this advice was reliable, and if so, personal experience trust values of these agents are increased; if not, personal experience trust values of these agents are decreased. Detailed formalization of this process will be given in Section III-C4.

Step 3D: If a majority consensus cannot be reached, then requiring majority consensus for advice is abandoned. Instead, the agent relies on role- and experience-based trust (e.g., taking the advice from the agent with highest role and highest experience trust value).[5]

Step 4: In order to eventually admit new agents into consideration, when advice is sought, the agent will ask a certain number of agents beyond agent a_n in the list. The responses here will not count toward the final decision, but will be scrutinized in order to update personal experience trust values, and some of these agents may make it into the top n list, in this way.

Algorithm 1 is a pseudocode summary of the proposed algorithm. Note that this pseudocode covers the main scenario, where an agent actively requests other agents for advice and does not include the exploration/testing step (see Step 4).

Algorithm 1: Computation Steps

```
while on the road do
    if in need of advice then
        Choose $n$; //number of agents to ask for advice
        //according to roles and experience
        Prioritize $n$ agents;
        Send request and receive responses;
        if response consensus > acceptable ratio then
            Follow advice in response;
        else
            Follow advice of agent with highest role and
            highest trust value;
    Verify reliability of advice;
    Update agents' trust values;
```

3) Role-Based Trust: Our role-based trust exploits certain predefined roles assigned to all agents in the system. The underlying assumption here is that the agents identified by authorities are more closely monitored and are expected to behave in a certain way. We can also conceptualize roles as an expected behavior of a certain group or class of agents, where all the agents belonging to a group would behave similarly. We propose a role-based approach because the expected number of possible roles and the rules to assign these roles would be very few in the domain of VANETs, and thus, can be manually managed and/or updated by a trusted authority. Note that the concept of seniority (expertise in a certain context/task, for instance) could be incorporated into role-based trust, as mentioned in Section III-B3.

To demonstrate our role-based approach, let us consider a simple system that recognizes the following four different roles listed in decreasing order[6], i.e., from the highest role to the lowest one: 1) authority; 2) expert; 3) seniority; and 4) ordinary. Each role level may also be associated with a trust value $T_r \in (0, 1)$, where higher level roles have larger T_r values. The rules for assigning and authenticating these roles can be structured as follows.

1) Agents representing authorities, such as traffic patrols, law enforcement, state or municipal police, etc., assume the authority role.
2) Agents specialized in road condition related issues, such as media (TV, radio, or newspaper) traffic reporters, government licensed, and certified instructors of driving school, etc., receive the expert role.
3) Agents familiar with the traffic or road conditions of the area in consideration, e.g., local people, who commute to work on certain roads or highways or have many years of driving experience with a good driving record (e.g., taxi drivers), are given the seniority role.
4) All other agents are considered having the ordinary role.

All agents should possess certificates issued by a trusted certificate authority for identification purposes. Note that we need

[3] For example, the number of agents available to ask in total is known, each agent could establish a preference for asking a certain percentage of those agents, when a large number is desired.

[4] There is no need for each group to have the same number of elements. We provide here only a simplified example.

[5] Note that an additional motive for modeling the trustworthiness of a variety of agents is to be able to learn about these agents for future interactions, for example, in the calculations of experience-based trust and majority-opinion trust.

[6] Our experience-based trust may be helpful for role categorization. When agents have sufficient experience-based trust information about each other, they may report this information to a trusted authority (i.e., the transportation department of government). A mapping between agents' real-world profiles and their trustworthiness can then be derived for helping categorize their roles.

a way for an agent to tell if another agent is indeed having the role that he is claiming to have. One possible solution to this problem is to make use of public key certificates in an asymmetric cryptosystem as follows. Each agent should have a public key certificate, which can simply be a document containing the agent's name, his role, and his public key. That document is signed by a trusted certificate authority (with the certificate authority's private key) to become the agent's public key certificate. Everyone can verify the authority's signature by using the authority's public key. Now, when agent A sends a message to agent B, A must sign the message with his private key. B then can verify (using A's public key) that the message was truly sent by A. Alternatively, a central authority (trusted third party) can be used to verify the agent's role and to authenticate trust, for agent B.

4) Experience-Based Trust: We track experience-based trust for all agents in the system, which is updated over time, depending on the agent's satisfaction with the advice given, when asked. As mentioned in the previous section, our experience-based trust is cumulative in the sense that it updates agents' trust recursively. Thus, only the most recent trust values and the number of interactions between agents are needed to be stored in the system, to make the system scalable. We here formalize the computation of this trust.

If we define the range of all personal experience trust values to be in the interval $(-1, 1)$, where 1 represents absolute trust and -1 represents absolute distrust, then we can use the following scheme to update an agent's personal experience trust value, as suggested by [14].

Let $T_A(B) \in (-1, 1)$ be the trust value indicating the extent to which agent A trusts (or distrusts) agent B according to A's personal experience in interacting with B. After A follows an advice of B, if the advice is evaluated as reliable, then the trust value $T_A(B)$ is increased by

$$T_A(B) \leftarrow \begin{cases} T_A(B) + \alpha(1 - T_A(B)), & \text{if } T_A(B) \geq 0 \\ T_A(B) + \alpha(1 + T_A(B)), & \text{if } T_A(B) < 0 \end{cases} \quad (1)$$

where $0 < \alpha < 1$ is a positive increment factor.

Otherwise, if B's advice is evaluated as unreliable, then $T_A(B)$ is decreased by

$$T_A(B) \leftarrow \begin{cases} T_A(B) + \beta(1 - T_A(B)), & \text{if } T_A(B) \geq 0 \\ T_A(B) + \beta(1 + T_A(B)), & \text{if } T_A(B) < 0 \end{cases} \quad (2)$$

where $-1 < \beta < 0$ is a negative decrement factor.

The absolute values of α and β are dependent on several factors because of the dynamics of the environment, such as the data sparsity situation mentioned in Section III-B2 and the event/task-specific property mentioned in Section III-B3. For example, when interaction data is sparse, these values should be set to be larger, giving more weights to the available data. For life-critical events (i.e., collision avoidance), $|\alpha|$ and $|\beta|$ should be larger, in order to increase or decrease trust values of reporting agents more rapidly. Also note that we may set $|\beta| > |\alpha|$ by having $|\beta| = \mu|\alpha|$ and $\mu > 1$ to implement the common assumption that trust should be difficult to build up, but easy to tear down. Setting α too generously possibly results in being too trusting of certain agents. Setting β too harshly may result in reducing the number of agents being trusted. In certain environments, we may need to be very defensive; this, however, is not always the case. We should be able to learn, through experience, whether we need to adjust the values set for α and β (i.e., we are being too generous or too harsh).[7]

We also incorporate a forgetting factor λ $(0 < \lambda < 1)$ in (1) and (2), allowing A to assign less weight to older interactions with B. This is to cope with the possible changes of B's behavior over time. If we define t as the time difference between the current interaction and the previous one,[8] the equations then become

$$T \leftarrow \begin{cases} \lambda^t(1 - \alpha)T + \alpha, & \text{if } T \geq 0 \\ \lambda^{-t}(1 + \alpha)T + \alpha, & \text{if } T < 0 \end{cases} \quad (3)$$

$$T \leftarrow \begin{cases} \lambda^t(1 - \beta)T + \beta, & \text{if } T \geq 0 \\ \lambda^{-t}(1 + \beta)T + \beta, & \text{if } T < 0 \end{cases} \quad (4)$$

where we substitute $T_A(B)$ by T for the purpose of clarity. The trust values A has of B will increase/decrease more slowly than those in (1) and (2) because older interactions between them are discounted and have less impact on the current trust values.

The number of interactions between agents A and B, $N_A(B)$, should also be discounted accordingly. This can also be done recursively as follows:

$$N_A(B) = \lambda^t N_A(B) + 1. \quad (5)$$

Note that the experience-based formulas are also valuable to cope with agents, who try to build up trust, and then, deceive. Once deception is detected, trust can be torn down quite quickly. Note that penalizing dishonesty more severely also acts as a disincentive for agents to simply gather reports from others and report them, in an effort to boost their trustworthiness. Since it is possible this information may be inaccurate, this strategy runs the risk of severely destroying trustworthiness.

5) Majority Opinion and Confidence: Suppose agent A in VANET receives a set of m reports $\mathcal{R} = \{R_1, R_2, \ldots, R_m\}$ from a set of n other agents $\mathcal{B} = \{B_1, B_2, \ldots, B_n\}$ regarding an event. Agent A will consider more heavily the reports sent by agents that have higher level roles and larger experience-based trust values. When performing majority-based process, we also take into account the location closeness between the reporting agent and the reported event, and the closeness between the time when the event has taken place and that of receiving the report. We define C_t (time closeness), C_l (location closeness), T_e (experience-based trust), and T_r (role-based trust). Note that all these parameters belong to the interval $(0, 1)$ except that T_e needs to be scaled to fit within this interval.

We denote $\mathcal{B}(R_j)$ as a set of agents reporting a same report $R_j \in \mathcal{R}$ $(1 \leq j \leq m)$, and $\mathcal{B}(R_j) \subseteq \mathcal{B}$. For each agent B_i $(1 \leq i \leq n$ and $m \leq n)$ belonging to $\mathcal{B}(R_j)$, we aggregate the effect of its report according to the aforementioned factors. The

[7] For an agent to precisely set α and β, further insight may be gained from [14], which provides proofs for avoiding infinite harm, for the case of trust modeling in e-marketplaces.

[8] The value of t may be scaled within the range of $[0, 1]$. This can be achieved by setting a threshold $t_{\max}$ of the maximum time for an agent to totally forget the experience happened at the time, i.e., $t_{\max}$ prior to the current time.

aggregated effect $E(R_j)$ from reports sent by agents in $\mathcal{B}(R_j)$ can be formulated as follows:

$$E(R_j) = \sum_{B_i \in \mathcal{B}(R_j)} \frac{\max(T_e(B_i), T_r(B_i))}{C_t(R_j)C_l(B_i)}. \tag{6}$$

In this equation, experience- and role-based trust are discounted based on the two factors of time and location closeness. The summation is used to provide the aggregated effect of the reporting of the agents.

Note that location closeness $C_l(B_i)$ depends only on the location of agent B_i. It can be formulated as follows:

$$C_l(B_i) = \lambda'^{-\triangle d(B_i)} \tag{7}$$

where λ' is a discounting factor, and $\triangle d(B_i)$ is the distance between the location of agent B_i and the location of where the reported event has taken place. Time closeness $C_t(R_j)$ depends on the time of receiving the report R_j. It can be formulated as follows:

$$C_t(R_j) = \lambda'^{-\triangle t(R_j)} \tag{8}$$

where $\triangle t(R_j)$ is the interval between the time of receiving the report R_j and the time when the reported event has taken place. $C_t(R_j)$ can also be written as $C_t(B_i)$ because we can assume that each report is sent by an unique agent in possibly different time.

To consider the effect of all the different reports, the majority opinion is then

$$M(R_j) = \arg\max_{R_j \in \mathcal{R}} E(R_j) \tag{9}$$

which is the report that has the maximum effect, among all reports.

A majority consensus can be reached if

$$\frac{M(R_j)}{\sum_{R_j \in \mathcal{R}} E(R_j)} \geq 1 - \varepsilon \tag{10}$$

where $\varepsilon \in (0, 1)$ is set by agent A to represent the maximum error rate that A can accept. A majority consensus can be reached if the percentage of the majority opinion (the maximum effect among different reports) over all possible opinions is above the threshold set by agent A.

If the majority consensus is reached, the majority opinion is associated with a confidence measure. This measure takes into account the number of interactions taken for modeling experience-based trust values of reporting agents and the maximum accepted error rate ε. We define $N(R_j)$ as the average of the discounted number of interactions used to estimate experience-based trust values of the agents sending the majority report R_j calculated using (5). The Chernoff bound theorem [12] provides a bound for the probability that the estimation error of the majority opinion exceeds a threshold, given the number of interactions. The confidence of the majority opinion can thus be calculated as follows:

$$\gamma(R_j) = 1 - 2e^{-2N(R_j)\varepsilon^2}. \tag{11}$$

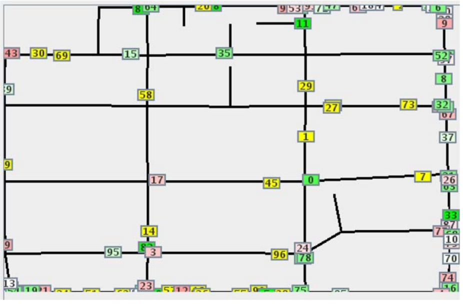


Fig. 2. Simulating VANET using SWANS simulator with STRAW mobility model.

IV. Experimental Evaluation

In this section, we present preliminary evaluation of our trust model. We use scalable wireless ad hoc network simulator (SWANS, jist.ece.cornell.edu) with street random waypoint (STRAW) mobility model [34]. SWANS is entirely implemented in Java and can simulate networks with potentially thousands of nodes (agents), while using incredibly small amount of memory and processing power. STRAW allows to simulate real-world traffic by using real maps with vehicular nodes that follow rules, such as speed limits, traffic signals, stop signs, etc.

We use a map of North Boston, MA. Fig. 2 shows a snapshot of one of our simulation runs. The bold lines are the extracted road segments from the map. The small rectangles labeled by integers represent vehicles running on the streets. For all our experiments, we fix the total number of vehicles to 100 and run the simulation for a total duration of 900 s of simulation framework time. Note that, in this paper, we only experiment with the role- and experienced-based dimensions of our trust model, while leaving more comprehensive experimental evaluation for future work.

A. Performance Metric

One of the applications of V2V communication is to be able to route traffic effectively through the VANET and to avoid congestion or hot spots. Malicious agents in the network may send untruthful traffic information, to mislead other agents, and cause traffic congestion. We measure the performance of our proposed trust model by observing to what extent it can cope with deceptive information sent by malicious agents. According to [34], we can measure congestion based on the average speed of vehicles. Lower average speed implies more traffic congestion. The performance of our model can then be measured as the increase in average speed of all agents by incorporating our model under the environment, where malicious agents exist.

B. Results

We present experimental results to clearly show the value of different trust metrics integrated in our expanded trust management and to demonstrate that the combined one is the most effective.

1) Effect of Liars on Average Speed: In our first experiment, we vary the percentage of malicious nodes in the environment

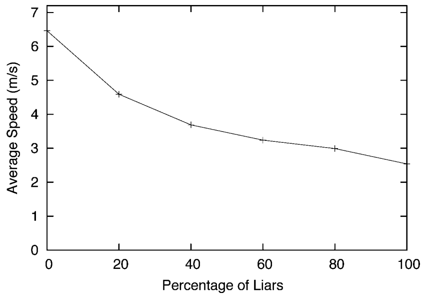


Fig. 3. Average speed of all cars when there are different percentages of liars.

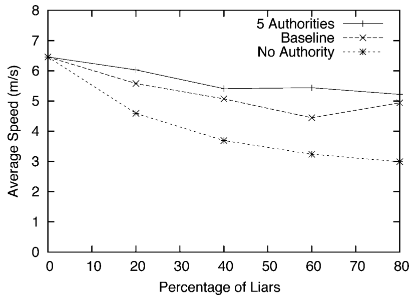


Fig. 4. Average speed of all cars with role-based trust.

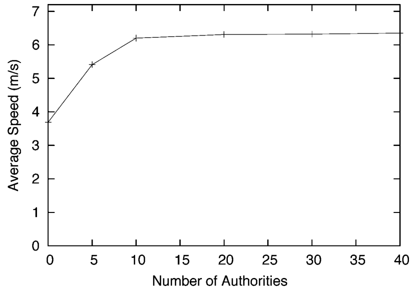


Fig. 5. Average speed of all cars when there are different numbers of authorities.

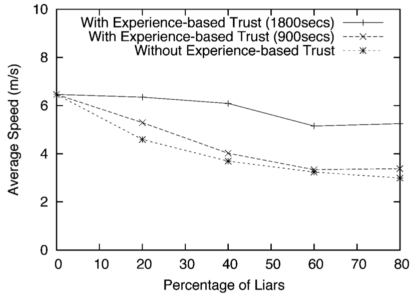


Fig. 6. Average speed of all cars with experience-based trust.

and measure the change in average speed of the vehicles in the network. We choose a lying strategy for the malicious nodes, where they always lie about congestion on a particular road segment, i.e., report congestion when there is no congestion and *vice versa*. We present the results in Fig. 3. As expected, average speed of vehicles in the network decreases as the percentage of liars increases.

2) Countering Liars With Role-Based Trust: Next we experiment with role-based trust, where we introduce some agents in the environment with the role of authorities, as mentioned in Section III-C3. In our simulation, authorities are assumed to be always trustworthy. In the first experiment, we present a baseline for role-based trust, where we use only the role-based dimension of our trust model, without majority opinion or priority. We fix the number of authorities to be five and vary the number of malicious agents from 0% to 80%. Results are presented in Fig. 4 labeled as *baseline*. The average speed of all cars is improved for any number of malicious agents as compared to the case, where there are no authorities present in the system. Next we conduct an experiment, where we use role-based trust along with priority and majority opinion. We fix the number of authorities to five. These results are also presented in Fig. 4 labeled as *five authorities*. We see a further improvement in the average speed of cars, for all cases, as compared to the baseline case, showing the effectiveness of role-based trust. This experiment also shows that if we have even a small number of agents with a role of authority in the system, we can still effectively cope with an increasing percentage of malicious nodes. In the next experiment, we fix the number of malicious agents to be 40%, and then, vary the number of agents with the role of authority between 0 and 40. These results are presented in Fig. 5. With an increase in the number of authorities in the environment, the overall average speed of the nodes increases, countering the effect of malicious agents. This further shows the effectiveness of role-based trust in our model.

3) Countering Liars With Experience-Based Trust: In this experiment, we employ only the experience-based dimension of trust. We vary the percentage of liars and measure the overall average speed of vehicles. For this experiment, we run the simulation for 900 and 1800 s. As we can see from Fig. 6, using experience-based trust results in an increase in the average speed of vehicles. This trend is consistent for all percentages of liars in the system, which shows that experience-based trust is able to cope with the lying behavior of malicious agents.

Note that for the 900-s case, the performance of our trust model, namely the speed of vehicles, is averaged over the total duration of only 900 s of the simulation framework time. At the beginning of the simulation, an agent does not yet have any experience with other agents. This explains the model's moderate performance during this early period. When we run the simulation for 1800 s, experience-based trust shows much better performance as compared to the 900-s case. This is to be expected because the longer we run the simulation, the more experience an agent has with other agents, thus it can more effectively cope with the lying behavior of other agents.

Fig. 7 shows the number of lying agents detected over the duration of a simulation using experience-based trust. In this experiment, the simulation is run for 900 s, and the number of liars is fixed to 40%. On the x-axis, we have the elapsed time in seconds, and on the y-axis, we have the number of liars detected. At the beginning of the simulation, an agent does not yet have any experience with other agents. This explains the relatively low number of detections during the time 0–100 s. As the simulation progresses, each agent gains more and more

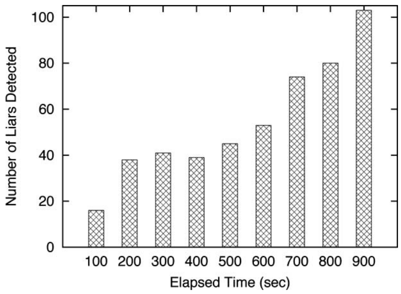


Fig. 7. Number of liars detected over time with experience-based trust.

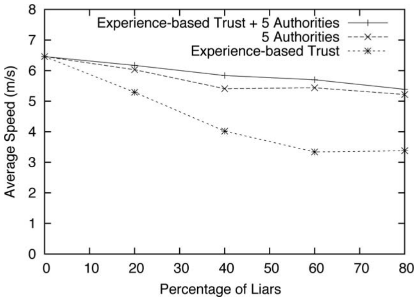


Fig. 8. Average speed of all cars with role- and experience-based trust.

experience, which results in an increasing number of detections over time, as shown in Fig. 7.

4) Combining Role- and Experience-Based Trust: From Figs. 4 and 6, we can see that even though experience-based trust results in an increase in the average speed of vehicles in the network with the presence of malicious agents, role-based trust does this job more effectively. In this experiment, we combine both dimensions together and measure the average speed. These results are presented in Fig. 8. As we can see, by combining these two dimensions, we can achieve an average speed, which is higher than when using any one of these two dimensions individually. This shows that a trust model for agents in VANETs can greatly benefit by combining several dimensions of trust as proposed in this paper.

5) Coping With Sparsity: This experiment is carried out to demonstrate the property of our model in coping with the data sparsity problem. In this experiment, we involve 50 nodes and run the simulation for 300 s of simulation framework time. We reduce the ratio of communication between nodes. The available data for modeling the trustworthiness of nodes is more sparse when the communication ratio is lower. As can be seen from Fig. 9, the percentage of detecting malicious nodes decreases when the ratio of communication is reduced. By decreasing the value of β, the ability of detecting malicious nodes is increased dramatically.[9] This indicates that our model is able to cope with the data sparsity problem by changing the parameter β to adjust the weight of available data.

The role-based trust in our model is also able to cope with data sparsity. As shown in Fig. 8, with only the experience-based trust, the performance difference of our model between more and fewer liars is large. This difference is reduced when the role-based dimension is also used. The role-based trust reduces the impact of more liars, and therefore, is able to begin to cope with the data sparsity problem.

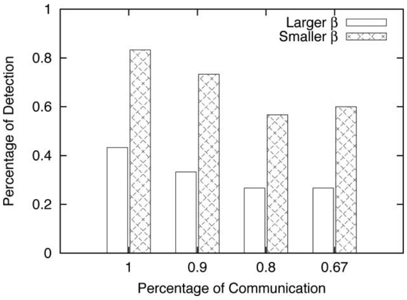


Fig. 9. Coping with sparsity.

V. Related Work

Lin *et al.* [35] have investigated the benefits achieved by self-interested agents in vehicular network through simulations. They consider a scenario, where agents can achieve road congestion information from other agents through gossiping. Two different behaviors of self-interested agents are investigated: 1) agents want to maximize their own utility and 2) agents want to cause disorder in the network. Simulation results indicate that for both behaviors, self-interested agents have only limited success in achieving their goals, even if no counter measures are taken. However, the authors realize the need to take these preliminary results to more complex and potentially more damaging scenarios that may arise in VANETs. They also identify the need to establish trust in VANETs through distributed reputation mechanisms, motivating our work.

In contrast to the traditional view of entity-level trust, Raya *et al.* [24] propose that data-centric trust may be more appropriate in the domain of ephemeral ad hoc networks, such as VANETs. Data-centric trust establishment deals with evaluating the trustworthiness of the data reported by other entities rather than trust of the entities themselves. In their model, they define various trust metrics of which *a priori* trust relationships in entities is just one of the default parameters and depends on the attributes associated with a particular type of node. Using Bayesian inference and Dempster–Shafer Theory, they evaluate various evidences regarding a particular event taking into account different trust metrics applicable in the context of a particular vehicular application. Finally, their decision logic outputs the level of trust that can be placed in the evaluated evidences indicating whether the event related with the data has taken place or not. There are some commonalities between our approach and theirs, for example, they also propose the use of task/event-specific trust metrics as well as time and location closeness. However, there are important differences as well. We combine these metrics in a fundamentally different way taking the traditional view of entity-level trust instead of data-centric trust. One of the shortcomings of their work is that trust relationships in entities can never be formed, only ephemeral trust in data is

[9] The absolute value of β in (2) reflects the weight placed on available data. Since $-1 < \beta < 0$, decreasing the value of β will increase its absolute value, and the weight of data will also be increased.

established, and because this is based on a per event basis, it needs to be established again and again for every event. This will work so long as there is enough evidence either in support of or against a specific event, but in case of data sparsity, we believe our model would perform better. We leave a detailed comparison between these two models for future work.

Dotzer [23] has suggested building a distributed reputation model that exploits a notion called opinion piggybacking, where each forwarding agent (of the message regarding an event) appends its own opinion about the trustworthiness of the data. They provide an algorithm that allows an agent 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 geosituation-oriented reputation level. This last trust metric allows their model to introduce some amount of dynamism in the calculation of trust by considering the relative location of the information reporting and the receiving nodes. Additionally, the situation-oriented reputation level allows a node to consider certain situational factors, e.g., familiarity with the area, rural, or metropolitan area, etc., again introducing some dynamism in trust evaluation based on context. Our model has direct trust in the form of experience-based trust, indirect trust in the form of role-based trust. Furthermore, we also use location closeness in our model that is similar to geosituation-oriented reputation level in their model. However, we provide an algorithm to combine, for example, experience- and role-based trust into a priority-based trust, at the same time taking the majority opinion into account. This way of combining these different metrics is a novel feature of our model and is tailored, specifically, for the domain of VANET. Additionally, our model does not rely on introducing opinion piggybacking in message passing and the associated algorithms to generate and aggregate opinions at each individual node.

Golle *et al.* [31] present a technique that aims to address the problem of detecting and correcting malicious data in VANETs. The key assumption of their approach is in maintaining a model of VANET at every node. This model contains all the knowledge that a particular node has about the VANET. Incoming information can then be evaluated against the agent's model of VANET. If all the data received agrees with the model with a high probability, then the agent accepts the validity of the data. However, in the case of receiving data, which is inconsistent with the model, the agent relies on a heuristic that tries to restore consistency by finding the simplest explanation possible and also ranks various explanations. The data that is consistent with the highest ranking explanation(s) is then accepted by the node. The major strength of this approach is that it provides strong security against adversaries that might even be highly trusted members in the network or might be colluding together to spread malicious data. The approach that we present in this paper is orthogonal to their approach. In particular, we do not aim to detect and correct malicious data in the network, instead we want to detect the entities (agents or cars) that are generating this malicious data, establishing trust, or distrust in the entity itself. This allows an agent to avoid an interaction with a distrustful agent in future.

Gerlach [36] has outlined a sociological trust model based on the principle of trust and confidence tagging. They have identified various forms of trust including situational trust—which depends on situation only, dispositional trust—which is the level of trust based on an agent's own beliefs, system trust—depends on the system, and finally, belief formation process—which is the evaluation of data based on previous factors. Additionally, they have presented an architecture for securing vehicular communication and a model for preserving location privacy of the vehicle. Again, though this model has some similar components as our model, for example, situational trust can be compared with event/task-specific trust, similarly dispositional trust can be compared to experience- or role-based trust. However, as opposed to our model, they place much emphasis on the use of their trust and security framework for developing context-aware applications and services for vehicular networks, taking into account the ease of development of such applications. Furthermore, their model focuses more on the system level security features, such as preserving location privacy.

A number of researchers have proposed trust and reputation models with role-based approach and the notion of confidence [22]. In particular, [37] introduced FIRE, a framework that integrates direct trust and role-based trust, in which the direct trust model of [17] is proposed as the method for capturing this element of the overall calculation, with some adjustment to consider, more carefully, the decay of trust values over time. In contrast, our model incorporates role- and experience-based trust, which are combined using a priority-based approach, together with majority-based trust to evaluate the trustworthiness of agents in the aggregate, while taking into consideration the important properties specific to VANET environments.

Wang and Singh [38] have further extended the notion of confidence to a certainty measure that takes into account not only the number of interactions but also the conflict among the reports of multiple reporting agents. Certainty decreases when conflict among reports increases, which is similar to our majority-based trust. In our majority-based trust, a majority consensus can be reached only when a significant majority agrees in the reports or little conflict exists among the reports. In this case, the asking agent will be certain and confident enough to follow the majority's advice.

VI. Conclusion and Future Work

The question of placing trust in the data received from other agents in VANETs can potentially become a question of life and death. The success of deploying VANETs, therefore, is contingent upon the success in establishing effective methods of trust establishment [35]. In this paper, we started by discussing some of the key challenges to modeling the trust of agents in VANET environments followed by identifying the areas, where the existing trust models in the domain of multiagent systems are lacking in their applicability to VANETs. We then presented our expanded trust model for agents in VANETs. Our model is a novel integration of several trust metrics, including role-, experience-, priority-, and majority-based trust.

To emphasize the contributions of our research, we note the following. In surveying related work, we highlighted the value of considering role-based trust [22], event-specific trust sensitive to time and location [24], and the tracking of confidence in a source of information [38]. Yet, researchers have not explored sufficiently well how to combine these elements into one comprehensive system for modeling trust in the domain of VANETs. What we have introduced in our research is the important concept of priority-based trust, which provides for an ordering of the value of an information source within a role category, using the valuable influence of experience-based trust. We have also advocated a limit on the number of sources consulted, to be sensitive to the task at hand. In addition, we have explicitly integrated our treatment of time and location considerations into a procedure for gauging whether majority consensus has been reached, which ultimately determines the advice an agent should follow. We have, moreover, considered the case, where the majority consensus is not apparent, to then rely on other elements of the trust modeling. The discussion of cases is included, where untruthful advice is detected due to the consideration of multiple facets of trust, at once. The various elements are important to consider together is further confirmed through our experimental results.

For future work, we plan to explore various extensions to our current model. One interesting topic to explore is how to make use of a "commuter pool"—a set of agents that travel the same route with some regularity, as mentioned in Section III-B2. This would provide a social network, where trust may be built up and frequent encounters may occur. This scenario would heighten the value of experience-based trust as part of the model.

Considering effective modeling of location information could also form an important thread for future research, due to its role in the calculation of majority-based opinion. For example, to avoid spoofing of location information, independent methods for vehicle tracking may need to be incorporated. We may also explore how to integrate incentives for drivers to opt into honest location reporting (e.g., as a precondition to receiving information from other vehicles).

To cope with various malicious attacks, in general, is another interesting topic of research. Collusion is notoriously difficult to address, but individual vehicles that are misreporting may possibly be detected due to differences with other vehicles, through our majority opinion algorithm. Our approach of combining experience-based, role-based, and majority-opinion trust offers some important checks and balances against dishonest reporting. For example, an agent may choose to constantly report that the roads are congested, assuming that agents receiving these reports will avoid the roads, and then, never discover the dishonesty. But the agent will be gathering multiple reports and if the majority opinion suggests that the road is free, the dishonest agent will be discovered. Likewise, high authority agents may exist to strongly discount the dishonest agent's reports.

The case, where agents fail to report events is also an interesting one to explore, for future research. If location-tracking information becomes more prevalent, failure to report a life-critical event at that location may be independent reason to decrease trustworthiness; vehicles in special roles (such as police) would likely serve to confirm the presence of such a life-critical event. Current models of trust and reputation in multiagent systems have focused more on evaluating the trustworthiness of information that has been received, rather than considering the lack of reporting. Perhaps some new ground in trust modeling would be introduced by this research.

For future work, we also plan to expand our experimental evaluation to include more complex scenarios, where we test the effectiveness of other components including *event/task* and *location/time* specific components. Approaches such as that of [39] or of [40] may be particularly valuable to consider, as they propose methods to also be context-sensitive when modeling multidimensional trust. Furthermore, it is also important to confirm the scalability of our trust model with an increasing number of agents in the system. In fact, increasing the number of vehicles in our simulations may also provide additional insights into how best to set the value of n in Step 1 of our algorithm.

We could also consider a scenario, where more than one agent (vehicle) in VANET forms a coalition with other agents to achieve a common goal. For instance, one such goal could be to cause mayhem in the network, which can be attributed to vandalism or terrorism [35]. The consequences can be very critical and might end up claiming many lives. Future experimentation could also include cases, where life-critical events, such as accidents are at play. In these cases, some kind of authority should be involved and this can serve to keep the other vehicles on the road honest in their reporting. A false report would differ with that of the authority. These experiments would, therefore, provide greater insights into the value of our concept of role-based trust.

As a final thread for future research, we may investigate the approaches of other authors, who are also concerned with the issues of scalability and privacy that we are interested in addressing within our model, in order to determine new directions. For example, a position-based clustering technique for communication between agents is proposed in [41]; this approach integrates positioning information in electing the head of a cluster in order to achieve better scalability. Preserving the privacy of an agent through the use of proxies in peer-to-peer data sharing has been explored in [42]. This work suggests that proxies may provide valuable masking of the identity of an agent, as long as they are trusted.

REFERENCES

[1] Department for Transport, "Reported road casualties great britain: 2008 annual report," Road Casualties Great Britain, U.K., 2008.

[2] *The Network on Wheels (NOW) Project*, (2006) [Online]. Available: http://www.network-on-wheels.de/

[3] *The Car-to-Car Communication Consortium (C2CC).* (2007) [Online]. Available: http://www.car-to-car.org/

[4] GM, *Threat Assessment Algorithm* (2004) [Online]. Available: http://www.nhtsa.dot.gov/people/injury/research/pub/acas/acasest/

[5] S. D. Ramchurn, D. Huynh, and N. R. Jennings, "Trust in multi-agent systems," *Knowl. Eng. Rev.*, vol. 19, no. 1, pp. 1–25, 2004.

[6] I. Chisalita and N. Shahmehri, "On the design of safety communication systems for vehicles," *IEEE Trans. Syst., Man, Cybern., Part A: Syst. Hum.*, vol. 37, no. 6, pp. 933–945, Nov. 2007.

[7] C. Leckie and R. Kotagiri, "Policies for sharing distributed probabilistic beliefs," in *Proc. ACSC*, 2003, pp. 285–290.

[8] S. Eichler, C. Schroth, and J. Eberspacher, "Car-to-car communication," in *Proc. VDE Congr.*—Innovations for Europe, Oct. 2006.

[9] K. Regan, P. Poupart, and R. Cohen, "Bayesian reputation modeling in e-marketplaces sensitive to subjectivity, deception and change," in *Proc. 21st Conf. Artif. Intell.*, 2006, pp. 1206–1212.

[10] J. Zhang and R. Cohen, "Trusting advice from other buyers in e-marketplaces: The problem of unfair ratings," in *Proc. 8th Int. Conf. Electron. Commerce*, 2006, pp. 225–234.

[11] D. J. Wu and Y. Sun, "The emergence of trust in multi-agent bidding: A computational approach," in *Proc. 34th Hawaii Int. Conf. Syst. Sci.*, 2001, vol. 1, pp. 1041–1048.

[12] L. Mui, M. Mohtashemi, and A. Halberstadt, "A computational model of trust and reputation," in *Proc. 35th Hawaii Int. Conf. Syst. Sci.*, 2002, pp. 2431–2439.

[13] S. Sen, "Reciprocity: A foundational principle for promoting cooperative behavior among self-interested agents," in *Proc. 2nd Int. Conf. Multi-Agent Syst.*, 1996, pp. 322–329.

[14] T. Tran, "A reliability modelling based strategy to avoid infinite harm from dishonest sellers in electronic marketplaces," *J. Bus. Technol., Spec. Issue Bus. Agents Semantic Web*, vol. 1, no. 1, pp. 69–76, 2005.

[15] R. Mukherjee, B. Banerjee, and S. Sen, "Learning mutual trust," in *Trust in Cyber-Societies*. New York: Springer-Verlag, 2001, pp. 145–158.

[16] D. Huynh, N. Jennings, and N. Shadbolt, "Developing an integrated trust and reputation model for open multi-agent systems," in *Proc. 5th Int. Conf. Auton. Agents Workshop Trust Agent Soc.*, 2004, pp. 65–74.

[17] J. Sabater and C. Sierra, "Regret: A reputation model for gregarious societies," in *Proc. 5th Int. Conf. Auton. Agents Workshop Deception, Fraud Trust Agent Soc.*, 2001, pp. 61–69.

[18] Y. Wang and J. Vassileva, "Bayesian network-based trust model," in *Proc. 6th Int. Workshop Trust, Privacy, Deception Fraud Agent Syst.*, 2003, pp. 372–378.

[19] Y. Wang and J. Vassileva, "Trust-based community formation in peer-to-peer file sharing networks," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell.*, 2004, pp. 341–348.

[20] B. Yu and M. P. Singh, "A social mechanism of reputation management in electronic communities," in *Proc. 4th Int. Workshop Cooperative Inf. Agents*, 2000, pp. 154–165.

[21] Y. Mass and O. Shehory, "Distributed trust in open multi-agent systems," in *Trust in Cyber-Societies*. Berlin, Germany: Springer-Verlag, pp. 159–173, 2001.

[22] W. Teacy, J. Patel, N. R. Jennings, and M. Luck, "Travos: Trust and reputation in the context of inaccurate information sources," *Auton. Agent Multi-Agent Syst.*, vol. 12, pp. 183–198, 2006.

[23] F. Dotzer, "VARS: A vehicle ad-hoc network reputation system," in *Proc. 6th IEEE Int. Symp. World Wireless Mobile Multimedia Netw.*, 2005, pp. 454–456.

[24] M. Raya, P. Papadimitratos, V. Gligor, and J. Hubaux, "On data-centric trust establishment in ephemeral ad hoc networks," École Polytech. Fédérale de Lausanne, Lausanne, Switzerland, Tech. Rep. LCA-REPORT-2007-003, 2007.

[25] B. Yu and M. Singh, "Searching social networks," in *Proc. Int. Auton. Agents Multi Agent Syst.*, 2003, pp. 65–72.

[26] G. Zacharia and P. Maes, "Trust through reputation mechanisms," *Appl. Artif. Intell.*, vol. 14, pp. 881–907, 2000.

[27] S. Rahman and U. Hengartner, "Secure vehicle crash reporting in vehicular ad hoc networks," in *Proc. 3rd Int. Conf. Secur. Privacy Commun. Netw. Workshops*, 2007, pp. 443–452.

[28] B. Yu and M. Singh, "Distributed reputation management for electronic commerce," *Comput. Intell.*, vol. 18, no. 4, pp. 535–549, 2002.

[29] E. Staab, V. Fusenig, and T. Engel, "Towards trust-based acquisition of unverifiable information," in *Cooperative Information Agents XII*. ser. LNCS, M. Klusch, M. Pechoucek, and A. Polleres, Eds. vol. 5180, New York: Springer-Verlag, 2008, pp. 41–54.

[30] C. Fung, J. Zhang, I. Aib, and R. Boutaba, "Robust and scalable trust management for collaborative intrusion detection," in *Proc. 11th IFIP/IEEE Int. Symp. Integr. Netw. Manage.*, 2009, pp. 33–40.

[31] P. Golle, D. Greene, and J. Staddon, "Detecting and correcting malicious data in vanets," in *Proc. 1st ACM Int. Workshop Veh. Ad Hoc Netw.*, 2004, pp. 29–37.

[32] A. Jøsang and R. Ismail, "The beta reputation system," in *Proc. 15th Bled Electron. Commerce Conf.*, 2002, pp. 324–337.

[33] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *J. Comput. Secur.*, vol. 15, pp. 39–68, 2007.

[34] D. R. Choffnes and F. E. Bustamante, "An integrated mobility and traffic model for vehicular wireless networks," in *Proc. 2nd ACM Int. Workshop Veh. Ad Hoc Netw.*, 2005, pp. 69–78.

[35] R. Lin, S. Kraus, and Y. Shavitt, "On the benefit of cheating by self-interested agents in vehicular networks," in *Proc. Int. Auton. Agents Multi Agent Syst.*. Honolulu: Hawaii, 2007.

[36] M. Gerlach, "Trust for vehicular applications," in *Proc. 8th Int. Symp. Auton. Decentralized Syst.*, 2007, pp. 295–304.

[37] T. Huynh, N. Jennings, and N. Shadbolt, "An integrated trust and reputation model for open multi-agent systems," *Auton. Agents Multi-Agent Syst.*, vol. 13, pp. 119–154, 2006.

[38] Y. Wang and M. P. Singh, "Formal trust model for multiagent systems," in *Proc. 20th Int. Joint Conf. Artif. Intell.*, 2007, pp. 1551–1556.

[39] A. Rettinger, M. Nickles, and V. Tresp, "Learning initial trust among interacting agents," in *Proc. 11th Int. Workshop Cooperative Inf. Agents*, 2007, pp. 313–327.

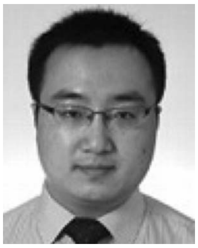
[40] M. Rehak and M. Pechoucek, "Trust modeling with context representation and generalized identities," in *Proc. 11th Int. Workshop Cooperative Inf. Agents*, 2007, pp. 298–312.

[41] Z. Wang, L. Liu, M. Zhou, and N. Ansari, "A position-based clustering technique for ad hoc intervehicle communication," *IEEE Trans. Syst., Man, Cybern., Part C: Appl. Rev.*, vol. 38, no. 2, pp. 201–208, Mar. 2008.

[42] Y. Lu, W. Wang, B. Bhargava, and D. Xu, "Trust-based privacy preservation for peer-to-peer data sharing," *IEEE Trans. Syst. Man, Cybern., Part A: Syst. Hum.*, vol. 36, no. 3, pp. 498–502, May 2006.

Umar Farooq Minhas received the Masters of Mathematics (MMATH) degree in computer science from the University of Waterloo, Waterloo, Canada, in 2008, where he is currently working toward the Ph.D. degree.

His current research interests include database systems, virtualization, performance modeling and analysis, highly available and fault-tolerant systems, cloud computing, multiagent systems, trust modeling, and vehicular ad hoc networks.



Jie Zhang received the Ph.D. degree from the University of Waterloo, Waterloo, Canada, in 2009.

He is currently an Assistant Professor at the School of Computer Engineering, Nanyang Technological University, Singapore. His research interests include artificial intelligence and multiagent systems, trust modeling and incentive mechanisms, and mobile and vehicular ad hoc networks.

Thomas Tran received the Ph.D. degree from the University of Waterloo, Waterloo, Canada, in 2004.

He is currently an Associate Professor at the School of Information Technology and Engineering, University of Ottawa, Ottawa, Canada. His research interests include artificial intelligence, electronic commerce, intelligent agents and multiagent systems, trust and reputation modeling, reinforcement learning, recommender systems, knowledge-based systems, architecture for mobile e-business, and vehicular ad hoc networks.

Robin Cohen received the Ph.D. degree from the University of Toronto, Toronto, Canada in 1983.

She is currently a Professor at the Cheriton School of Computer Science, University of Waterloo, Waterloo, Canada with research interests in multiagent trust modeling, user modeling and intelligent interaction.