Managing Social Networks in Vehicular Networks Using Trust Rules

Noor Abbani, Mohamad Jomaa, Takwa Tarhini, Hassan Artail, Wassim El-Hajj
Department of Electrical and Computer Engineering
American University of Beirut
Beirut, Lebanon
Emails: {nma51, mfj03, tat07, hartail, we07}@aub.edu.lb

Abstract—**Drivers and passengers in urban areas may spend large portion of their time waiting in their cars on the road while commuting to and from work, to school, or to the supermarket. Regularities of driving patterns in time and in space motivate the formation of communities of common backgrounds and interests. We propose a model for forming and maintaining Vehicular Social Networks (VSNs) that uses trust principles for admission to social groups, and controlling the interactions among members. This paper describes the details of the design, and proposes a simple but representative probabilistic model for deriving the probability of wrongful admissions and the probability of an agent trusting a malicious node. The experimental results, which were obtained from simulations using the network simulation software ns2, describe metrics related to the dynamics of group formation and time to form groups as well as to detecting malicious members. Our system was able to form social groups with agents of common interests and maintain an accurate trust evaluation of their behavior.**

Keywords-social networks; vehicular networks; social trust

I. INTRODUCTION

People always tend to communicate with each other, using different available means. However, people do not communicate randomly; instead, they do so in social units, in which participants usually share the same goals or interests. This is how the idea of social networking emerged; social networks are basically social structures where individuals share certain characteristics such as values, ideas, visions, friends and interests, making their interaction more focused [1]. An example of a social network would be commuting on the road; people usually take the same road everyday to get to their job or school, and often get stuck in traffic at the same place at the same time on a daily basis. Naturally, traffic makes most people frustrated and angry which affects their work productivity negatively and increases their stress level. A remedy for such problems is to allow vehicle occupants to interact and make use of their time while on the road via the creation of vehicular social networks (VSN), which are social networks formed on the roadways [2].

Although several existing applications exhibit some kind of mobile social interaction, none has considered vehicular networks as the underlying network that use a social network overlay. The major feature is that roadways provide a sufficient and regular concentration of people who may wish to socialize [2]. VSNs have numerous applications and can be used for emergency purposes, advertisement dissemination, professional referrals, meeting people, and entertainment.

In this work, we propose a framework for forming and managing such VSNs allowing vehicle occupants to engage in productive and/or entertaining activities through trust-aware communication with others. Our model is decoupled from the underlying physical VSN infrastructure, as it only requires equipping vehicles with wireless communication technologies.

Nevertheless, the communication within these social networks should be governed by the notion of *trust*, since the interactions between people have to be monitored by a mutual relation of trust in order for it to be satisfactory for all parties.

The work described in this paper contributes to the effort of developing effective and practical models for building and managing VSNs that are meant to allow vehicle occupants to exchange data and information in a trusted environment.

In the rest of this paper, section II provides a review of recent related research. Section III describes the system architecture. Section IV provides an analytical modeling of certain key performance indicators, while Section V describes the simulation setup and results. Finally, Section VI concludes the paper and presents the future work.

II. RELATED WORK

This section presents the most relevant work to VSNs. As will be evidenced throughout this paper, our work provides analytical and experimental evaluation of the proposed system's performance, whereas most of the works discussed in the literature only suggest VSN designs without supplying performance results. Moreover, compared to other similar systems, our system is more complete in that it describes and analyzes the overall system and its interactions.

In [2], A VSN system named RoadSpeak is presented. Its goal is to allow drivers to automatically join VSNs along common roadways, and interact with each other by means of voice chat messages. RoadSpeak follows the client-server model, which may not be optimal for a naturally decentralized environment, like that of vehicular networks.

Trust in such networks is crucial but is difficult to establish. A trust model named Situation-Aware Trust (SAT) was proposed in [3] to try to strengthen the tie between Internet infrastructures. SAT uses descriptive attribute-based cryptography to build trust policies and transform trust from Internet social communities to VANETs. The architecture was described without an implementation, which does not reveal information about its performance. Related to this is the work done in [4] that studied the applicability of using ordinary fixed networks to implement trust in VANETs. According to this work, trust establishment in VANETs involves infrastructure based trust and self organizing trust. The notion of trust is closely related to the presence of malicious nodes in the social network. The issue of dealing with such nodes is addressed in [5], which proposes handling malicious nodes based on different policies. Malicious nodes are defined as those that attack other nodes and influence the performance of the system. Most of the existing systems, according to [5], follow the same strategy in that once a malicious node is discovered the system will simply isolate it from the rest of the network.

A trust management model for VANETS that integrates cryptography-based entity trust and email-based social trust is proposed in [6]. This work is basically presenting research challenges without implementation. Another challenge that VANETs face according to [7] relates to forwarding event messages while ensuring that information is trusted by the receiving nodes. A solution relying on reputation is proposed.

III. System Architecture

We intend to integrate the social interaction in vehicular social networks with a trust management system that exploits the characteristics of such networks. In all, we expect our system to offer the functionalities listed below:

- Formation of groups based on common characteristics where nodes become members of several groups.
- Interaction between nodes where personal and group data, and recommendation requests and replies are exchanged.
- Trust management of node trust levels with respect to other nodes and to the groups it belongs to.
- Trust evaluation and update being a function of the nodes' behavior, interaction, activity and participation.
- Complete decentralization where group management and updates are automatically exchanged between nodes.
- Data exchange integrity where the flexibility in data exchange depends on the mutual trust between nodes.

The different modules that realize these functionalities are shown in the system architecture in Fig. 1.

A. Group Formation

Initially, each vehicle creates a group in which it is the only member. The vehicle user then broadcasts the group ID with his interests and hobbies; this allows nodes with common attributes to merge together resulting in the initial clustering of the cars into several social groups. This process takes place through the *Group Membership Handler* module. Later on, the members have the choice of sharing their private information (as age or profession) with other group members, or keeping them preserved.

B. Joining a Social Group

Nodes periodically send their neighbors their group ids and characteristics to advertise their groups. When a node is interested in a group, it sends to the advertising node a request to join it. When the other node receives this demand it has to send to the group members a vote request of their evaluation of the trustee. This is done by the *Admission Request Processor* module which then has to receive the votes and process them according to the rules that we describe later.

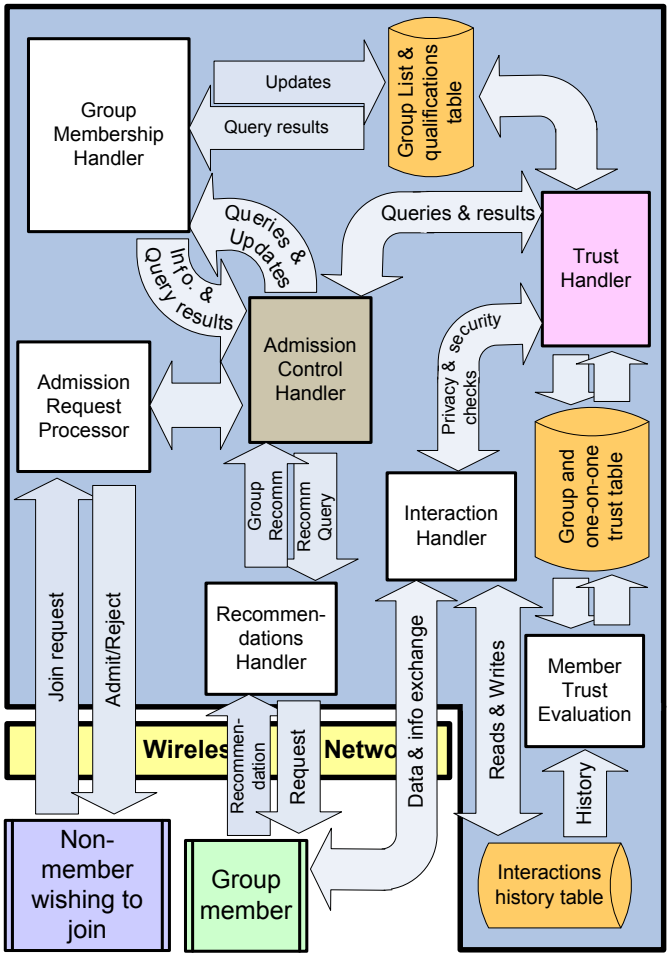

Figure 1: VSN Architecture

The votes are then sent to the *Recommendations Handler*, which saves the recommendation sent by every voter. This saved data should be monitored and checked on a constant basis and hence the presence of an *Admission Control Handler*

module. In the case of group related issues such as a member not replying to a vote request about a trustee, the node sends to all other group members a report that includes a complaint about that member. These reports are periodically checked, and when the number of complaints about a member exceeds a threshold, he gets fired from the group and all the members are notified. Figure 2 illustrates the process of joining a group.

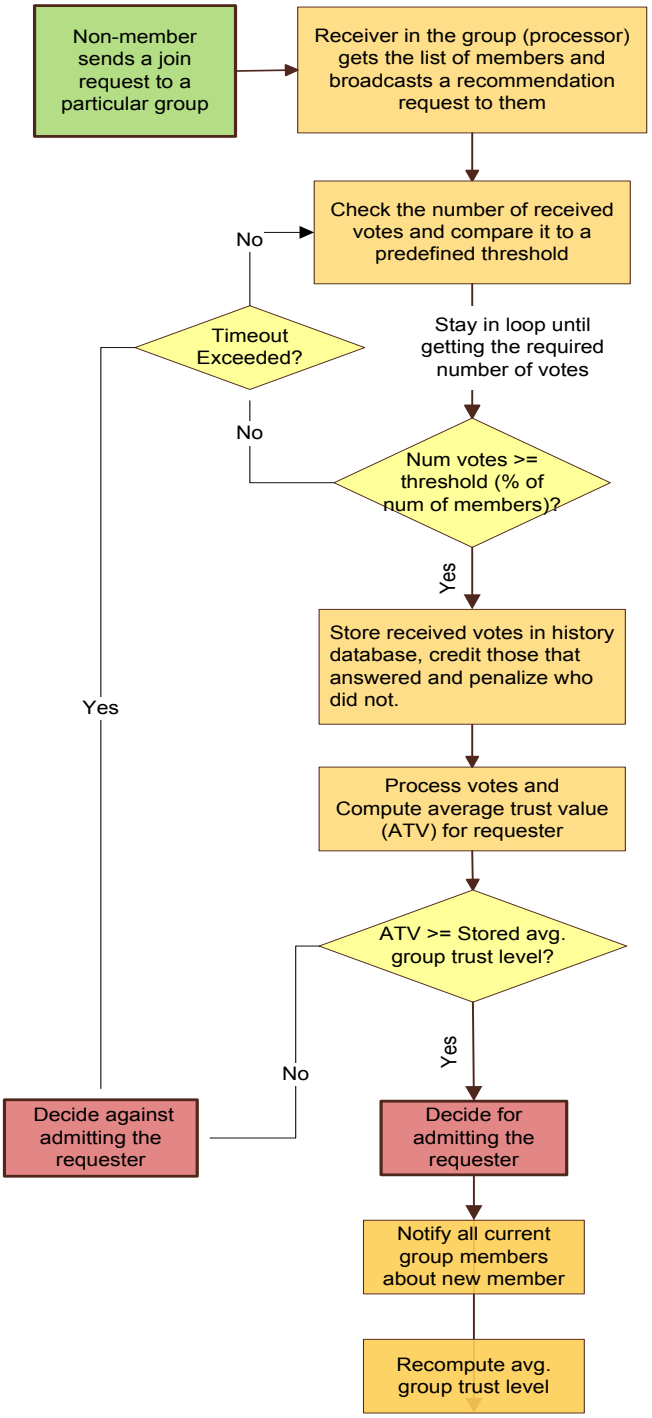

Figure 2: Joining group diagram

C. *Recommendation Process*

When a node receives a join-group request from a neighbor node (the trustee), it has to go through a recommendation process to decide whether to admit this trustee to the group or not. This is done by the social agent since the social groups do not even have group administrators, which allows for complete decentralization where central points of failure are avoided.

At this point it is important to state that there are two trust values for an agent: (1) one-on-one trust level with each other node it is interacting with, (2) trust level of a node in a group. All trust levels have values between 0 and 1.

In order to admit the trustee to the group, the trustee's trust level should be calculated and compared to the average trust level of the group. This is the average of all trust values of the group members. There are two features that contribute to the computation of this trust level: (1) matching interests between group and trustee, (2) recommendations given by group members about him. The computation equation therefore is:

$$J = \alpha \times Q + (1 - \alpha) \times \sum_{k=1}^{n} X_k R_k \quad (1)$$

where:

- α: weight given to the agent's qualifiers
- Q is the number of matching qualifiers (age, field, hobbies) between the node and the group.
- n: number of the group members
- R_k: trust level provided by the recommender to the group regarding the node requesting admission (using one-to-one trust value between the recommender and the trustee). This is a uniform random variable over the interval [0,1]
- X_k: trust level of the recommender in the group. This is also a uniform random variable in [0,1].

We define a new node in the network to be one that has not interacted with more than 70% of the group members before. If the node is not new, α is chosen to be 0.25; a higher significance of 75% is given to the recommendations of the group members. Otherwise, 75% is given to the matching qualifiers and the remaining 25% to the members' votes. This is to ensure that a new node will not be unfairly evaluated as malicious due to the null one-to-one trust values with the members. As for the recommendations, the processor (the node that received the join request) sends recommendation requests to all group members who reply with their one-to-one trust value with the trustee. This process stops whenever the processor receives a certain percentage of the replies. The remaining replies would be considered late, and their trust values would be negatively affected as will be explained later in the trust management section.

For the node to be admitted to a group, its computed trust value should be greater than the average. The decision is then sent to the trustee and to all members to update their group information. This process is illustrated in Figure 2.

D. *Trust Management*

The trust value of a particular node depends on its:

- Cooperativeness: the time it takes the agent to reply to a recommendation and whether it replies or not.
- Credibility: whether the agent gives correct data.

- Reliability: the accuracy of its recommendations
- Social value: communication between nodes of the same social group is more flexible.

To monitor these attributes, each social agent needs to store a history profile about the agents it is communicating with. This is done by a continuous monitoring of their behavior, accomplished in the *Trust Handler* of Fig. 1, and involving checking the trust level assigned by the recommender to a particular trustee over a period of time and comparing this voted trust value with the behavior of the trustee in question. For instance, suppose a malicious social agent, that tries to distort other nodes' reputation, receives a request to vote about a particular trustee. The processor, when performing the evaluation of the assigned recommendations, compares the received feedbacks about the trustee who had become a group member with his performance in the group. If the new member is a good agent, while some recommendations refer to him as malicious, the processor can infer that the recommender is a malicious agent. This causes the trust levels of the recommender to be negatively affected. His trust level in the group will also be decremented.

Additionally, there is tracking of late replies and the number of no replies. This will definitely affect the trust level of the group member and the one-on-one trust value with all the nodes the social agent is interacting with.

E. Trust Modeling

In this section, we present the equations that manage trust values in the network. The calculations are performed through the *Member Trust Evaluation*, which then writes the results to the *Group and One-on-one Trust Table* that will be used by the *Trust Handler* in guiding the interactions of the node.

The trust level between two nodes is updated in two cases:

1. Based on the reply time of the nodes:

During the exchange of data between two nodes, which takes place through the *Interaction Handler* in each of them, they both evaluate the reply time of each other, which will allow each node to update the trust values it gives to the other one. The trust level and the reply time are inversely proportional. The update happens according to the following formula (this formula applies for values of old Trust between 0.04 and 0.96 in order to make sure the value of trust is always positive and less than 1):

$$newTrust = (1 + R(replyrating)) \times oldTrust \quad (2)$$

Where R is a function of the reply time of the node in question, as follows:

$$R(replyrating) = \begin{cases} 0.04, & replyrating = very\ fast \\ 0.02, & replyrating = fast \\ 0, & replyrating = moderate \\ -0.02, & replyrating = slow \\ -0.04, & replyrating = very\ slow \end{cases} \quad (3)$$

This leaves it to the node itself to rate the reply time of other nodes and set its own rating standards based on which it evaluates the value of the variable "replyrating".

2. Based on the history of the nodes:

Each time a node A asks a node B for data or for recommendations about other nodes, it records whether node B replied or not, and records as well the information that B gave about the other nodes. The fraction of the times that B didn't reply is constantly checked by A, since it affects the trust value between them. Moreover, A computes with time a value for B evaluating it as a recommender; the computed value also affects the trust level.

These two ideas are reflected in the following formula:

$$newTrust = (1 - fracNoReplies + reclevel) \times oldTrust. \quad (4)$$

IV. MATHEMATICAL ANALYSIS

In this section, we mathematically evaluate the performance of our system in terms of Trust. Our aim is to minimize the "*Probability of Failure: F*" and the "*Probability of Wrongful Admission: W*", both of which we define below.

In our analysis, we use extensively the probability theory and random variables' properties. We refer to the central limit theorem (CLT) which *states that the sum of random variables is approximately normally distributed if the number of observations is large* [8]. The normal distribution is defined by 2 parameters, the mean μ and the standard deviationσ.

We now list the variables used in our analysis:

- **X**: Let X be the trust level given by a group to a certain node A when it is first admitted into the group (the group's average trust level). X is considered to be a normal random variable since it is the average of many other trust levels which are themselves random variables (CLT).
- **M**: Since A will definitely belong to many groups (recall that groups are formed according to hobbies and interests), we assume M to be the average of all the trust levels given to A by the various groups it belongs to. Consequently, M is a normal random variable by CLT.
- **T**: Let T be the trust value given by a node to another. T is a uniform random variable in the interval [0,1] since the trust value can equally likely take a value in this interval.
- **J**: J is the trust level computed by a group for a node based on the following formula:

$$J = 0.25 Q + 0.75 \sum_{k=1}^{n} X_k R_k \quad (5)$$

Note that Q being the number of matching qualifiers between the node and the group, we consider *y* independent qualifiers, each of which can be a match between the node and the group with probability 0.5. The success rate (Q) in the *y* independent trials is thus a Binomial random variable. J, on the other hand, is a Normal random variable by CLT.

We also use the following rules as guidelines for deciding whether a node is malicious or trustworthy:

- A node is malicious if $M \leq m^*$.
- A node is trusted by another node if $T \geq t^*$.

Our goal in this analysis is to find the optimal or near optimal values of $m^*$and t^*such that F and W are minimized.

I. Probability of failure: F

The probability of failure is the probability that a node A trusts a node B, with B being malicious. We approach the problem by first calculating the probability for a node to be malicious$P(M \leq m^*)$. We then calculate the probability for a node to be judged trustworthy by other nodes (regardless of being malicious or not)$P(T \geq t^*)$. F = $P(T \geq t^* \cap M \leq 0.3)$.

Aided by simulation results, we found for M: $\mu = 0.5$ and $\sigma = 0.289$. The probability for a node to be malicious:

$$P(M \leq m^*) = P\left(Z \leq \frac{m^*-0.5}{0.289}\right) \text{ [9]} \quad (6)$$

For m*< 0.5:

$$P(M \leq m^*) = P\left(Z \leq \frac{m^*-0.5}{0.289}\right) = 1 - P\left(Z \leq -\frac{m^*-0.5}{0.289}\right). \quad (7)$$

Z is the standard normal random variable. Using the standard normal distribution table [9], we show in the left graph of Figure 3the plot of $P(M \leq m^*)$ for all values of m^* such that$0.1 \leq m^* \leq 0.6$. The probability of a node being malicious clearly increases as m*increases. We want a value for m^*that would give a relatively low probability of maliciousness while being reasonably illustrative for a social network. We decided to go with $m^* = 0.3$ as it best simulates a vehicle social network.

Substituting m^* in the equation:

$$P(M \leq 0.3) = P\left(Z \leq \frac{0.3 - 0.5}{0.289}\right)$$
$$= 1 - P(Z \leq 0.69)$$
$$= 1 - 0.7549 = 0.2451.$$

Thus, a node is malicious with probability 0.2451.

Now we compute the probability for a node to be trusted by other nodes. For the uniform random variable T:

$$P(T \geq t^*) = 1 - P(T \leq t^*) = 1 - \int_0^{t^*} \frac{1}{1-0} dx = 1 - t^* \quad (8)$$

The probabilities calculated above ($P(M \leq 0.3)$ and $P(T \geq t^*)$) are considered independent in order to evaluate the nodes classification by our system as malicious or not. We do this by calculating the probability of a malicious node to be trusted:

$$P(T \geq t^* \cap M \leq 0.3) = P(T \geq t^*) \times P(M \leq 0.3)$$
$$= (1 - t^*) \times 0.2451. \quad (9)$$

So maximizing the value of t* minimizes the probability of failure F, but makes the communication between users harder. We needed a tradeoff between these 2 constraints; choosing t*=0.6 gives 9.8% error, which is reasonable keeping the communication among users relatively easy to initiate.

II. Probability of wrongful admission: W

W represents the probability for a group to admit a social agent while it should have rejected it. Let avg be the average trust level of the group members. Recall that J is the trust level computed by a group for a node, and that the node is admitted to a group if $J \geq avg$. J is a normal random variable that has (through simulation): μ=0.5 and σ = 0.289. The probability of admitting a node to a group becomes:

$$P(J \geq avg) = 1 - P(J \leq avg)$$
$$= 1 - P(Z \leq \frac{avg-0.5}{0.289}). \quad (10)$$

We plot this probability equation versus different values of avg ranging between 0 and 1 (right graph of Figure 3).

Aided by our simulation program, we found the average trust level of the various groups to always be in the range [0.55, 0.7]. Using the graph, this translates to a probability of admission in the range [0.2451, 0.4325]. Then:

$$W = P(J \geq avg \cap M \leq 0.3) = P(J \geq avg) \times P(M \leq 0.3)$$
$$= [0.2451, 0.4325] \times 0.2451$$
$$= [0.06, 0.106]$$

Hence, the probability of wrongful admission is between 6% and 10.6%, which is satisfactory for a social network.

It is clear from the analysis presented above that it is relatively easy for designers to change the values of m^* and t^*in a way that best fits their model.

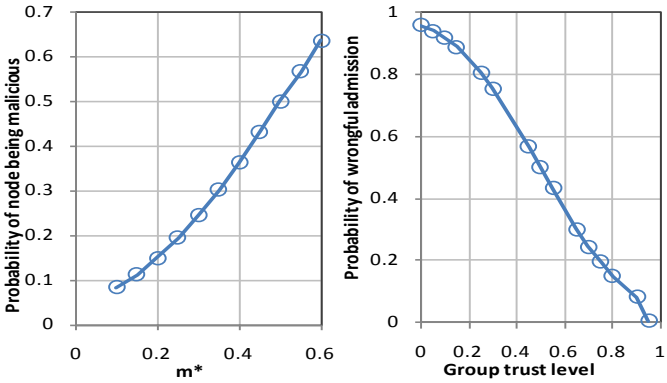

Figure3: Probability of malicious nodes and of wrongful admission

V. Experimental Evaluation

The simulations are performed in the network simulator NS-2. The system was tested for different scenarios in a 1km × 1km cluster of 30 to 100 nodes. These dimensions were dictated by the need of modeling car traffic on a specific road. The transmission range of each node is set to be 100 m. To simulate mobility, we used a mobility generator tool called *setdest* that randomly generates node movement.

The metrics that evaluate the performance of our system are:

1. Closeness in trust: measure of how close nodes in a particular group are in terms of their trust levels.

2. Closeness in interest: measure of how close nodes in a particular group are in terms of their interests.
3. Effective group formation: the dependence of number of groups formed on the initial distinct number of interests.
4. Latency: the time to detect a malicious node.

In this scenario, the nodes interact for 60 minutes and each node was given specific age, profession, field and hobbies. After 30 minutes, the nodes formed 12 distinct groups, each having a set of top interests; these are the interests that are most commonly shared by the group members. To evaluate the closeness in trust, we calculated the variance of the trust levels of all members in each group.

From Fig. 4, the maximum variance is 0.58, equivalent to a standard deviation of 0.23, showing that the trust levels in groups are close to each other. This illustrates the effective exploitation of social network characteristics. This also shows that malicious nodes get fired from the group after some time.

As for the closeness in interest, we computed 3 measures, one for each type considered (hobbies, profession, age). From Fig. 4, we can see that the highest value is 0.33 which shows that the groups formed consist of members with a high commonness in hobbies or profession.

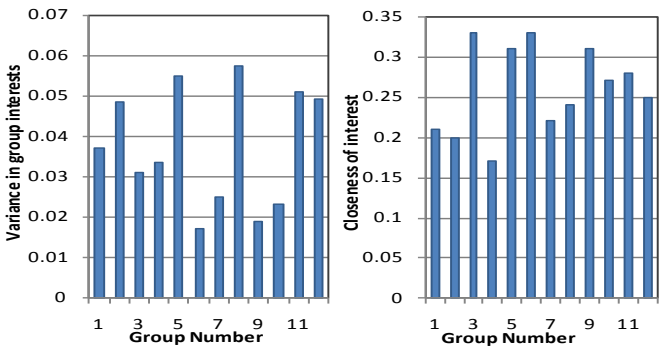

Figure 4: Variance of trust and closeness of interests in groups

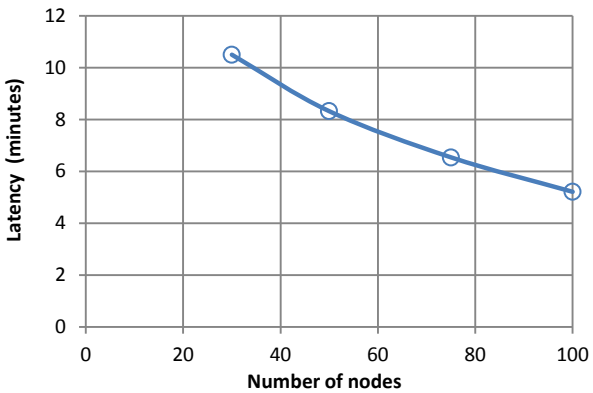

Figure 5: Latency versus number of nodes

To measure latency, we put 10 malicious nodes in the network (nodes that do not reply to recommendation requests or participate in activities). The simulation results indicated 9 detected and fired nodes while one node stayed in its group. The latency can be measured by the number of interactions that occurred with the malicious node until it is fired. From Fig. 5, we can see that the time to detect and fire a passive vehicle decreases as the network grows. This is due to the fact that the malicious node might be a member of more groups, which allows it to interact with more nodes, hence fired faster.

VI. CONCLUSION

In this paper, we proposed a novel framework for vehicle social networks with a dynamic trust capability that aims to minimize the impact of malicious behaviors in the social network. The presented mathematical analysis and simulation results validate the viability and potential of the proposed system. For future work, we intend to enhance our model to react dynamically to network characteristics and changes. We also plan to develop a more elaborate mathematical model for trusted VSNs, encouraged by the fact that future trends of on-board units installed in vehicles make systems like the one we proposed highly feasible to implement.

REFERENCES

[1] L. Sorensen, "User Managed Trust in Social Networking - Comparing Facebook, MySpace and LinkedIn", Wireless VITAE, 2009

[2] S. Smaldone, L. Han, P. Shankar, and L. Iftode,"RoadSpeak: Enabling Voice Chat on Roadways using Vehicular Social Networks", Proceedings of the 1st workshop on Social network systems,SocialNets '08,pp. 43-48, 2008

[3] X. Hong, D. Huang, M. Gerla, and Z. Cao, "SAT: Situation-Aware Trust Architecture for Vehicular Networks", 2008

[4] P. Wex,J. Breuer, A. Held, T. Uller, and L. Delgrossi, "Trust Issues for Vehicular Ad Hoc Networks", Proceedings of the IEEE Vehicular Technology Conference, 2008

[5] M. Cai, X. Wen, W. Zheng, Y. Cheng, andY. Sun, "Different-Strategy Management of Malicious Nodes in the Peer-to-Peer Network", International Conference on Environmental Science and Information Application Technology, 2009

[6] D.Huang, Z. Zhou, X. Hong, and M.Gerla, "Establishing Email-based Social Network Trust for Vehicular Networks", Proceedings of the 7th IEEE Consumer Communicationsand Networking Conference, 2010.

[7] F. Dotzer and L. Fischer, "VARS: A Vehicle Ad-Hoc Network Reputation System", Proceedings of the 6th IEEE Int'l Symposium on a World of Wireless Mobile and Multimedia Networks, 2005.

[8] Rice, John (1995), Mathematical Statistics and Data Analysis (Second ed.), Duxbury Press, ISBN 0-534-20934-3

[9] J. C. Turner, "Modern Applied Mathematics",English Universities Press 1970.