An Enhanced Community-based Reputation System for Vehicular Ad Hoc Networks

Hayoung Oh[1], Cliff C. Zou[2], Soyoung Park[3*]

[1]DASAN University College, Ajou University,
206 Worldcup-ro, Woncheon-Dong, Yeongtong-gu, Suwon, Korea, 443749
hyoh@ajou.ac.kr
[2]Dept. of Electrical Engineering and Computer Science, University of Central Florida,
4000 Central Florida Blvd. Orlando, FL, U.S.A, 32816-2362
czou@cs.ucf.edu
[3]Dept. of Internet and Multimedia Engineering, Konkuk University,
120 Neungdong-ro, Gwangjin-gu, Seoul, Korea, 05029
soyoungpark@konkuk.ac.kr

Abstract— **Reliable traffic message communication is one of the most essential goals for implementing secure vehicular ad hoc networking environment. In order to support secure traffic message communications, we propose an improved vehicular reputation system that manages the reputations of vehicles and verifies the reliability of traffic messages based on the vehicle reputation. We enhance our previously proposed community-based reputation model that an agent roadside unit (A-RSU) designated by a vehicle manages the reputation of the vehicle, and issues reputation certificates. In particular, we introduce practical ways of evaluating vehicle's reputation in real time and delivering the evaluated results to the A-RSU efficiently. The reputation of a vehicle is measured by the communication behaviors of the vehicle. All traffic event-based messages of the vehicle are evaluated by neighbors, and a local proof of the evaluation is delivered to the A-RSU. Finally, the A-RSU updates the reputation certificate of the vehicle based on the local proofs. We introduce an aggregated local proof generation mechanism based on the agreement of neighbors, and also provide efficient local proof delivery strategies. Our simulation results show that our proposed routing model outperforms in terms of average link duration, throughput and routing overheads than other routing strategies.**

Keyword- vehicular ad hoc networking, secure traffic message communications, community-based reputation model

I. Introduction

The Vehicular Ad hoc Network (VANET) is a special form of mobile ad-hoc network that allows communications among vehicles and infrastructural units. VANET can provide various types of applications such as emergency/safety warnings, cooperative driving, traffic-information sharing, Internet access, and location-aware advertising [1]-[2]. To activate these applications, however, secure and reliable traffic-data transmission must first be guaranteed; most of all, vehicle-generated traffic data that is transmitted to other vehicles should not be fabricated or forged during the transmission of a message, and vehicles should also be able to verify the validity of the received traffic data.

Over recent years, numerous works have been performed to implement a secure VANET. We also previously proposed a vehicular reputation system [3] whereby the reliable communication of traffic messages is a preliminary result. The purpose of a vehicular reputation system is the measurement of the reputation of each vehicle through the evaluation of each vehicle's communication behaviors. Since a vehicle's reputation shows the reliability of a vehicle that has been built up over a long time period, the traffic messages that are provided by highly reputed vehicles are likely to be more reliable.

We introduced a concept of the community-based reputation system [3] that exploits roadside units (RSU) to manage vehicles' reputations. This new concept exploits the finding that the majority of people drive their vehicles locally for their daily commute (to workplaces, schools, daycares, and superstores, etc.), and therefore, most vehicles have a predefined, constant daily driving trajectory, as illustrated in Figure 1. Among all of the smart vehicles that pass by an RSU, a substantial number of them will be repeatedly observed by the RSU on a daily basis. From the perspective of an RSU, the commuter vehicles that pass by it daily form a relatively stable

[*] Corresponding Author

"virtual community"; therefore, it is convenient and feasible to let the RSU take charge of managing the reputation scores of these vehicles. In the proposed reputation system, RSUs provide the reputation certificates of the vehicles that belong to their corresponding virtual communities.

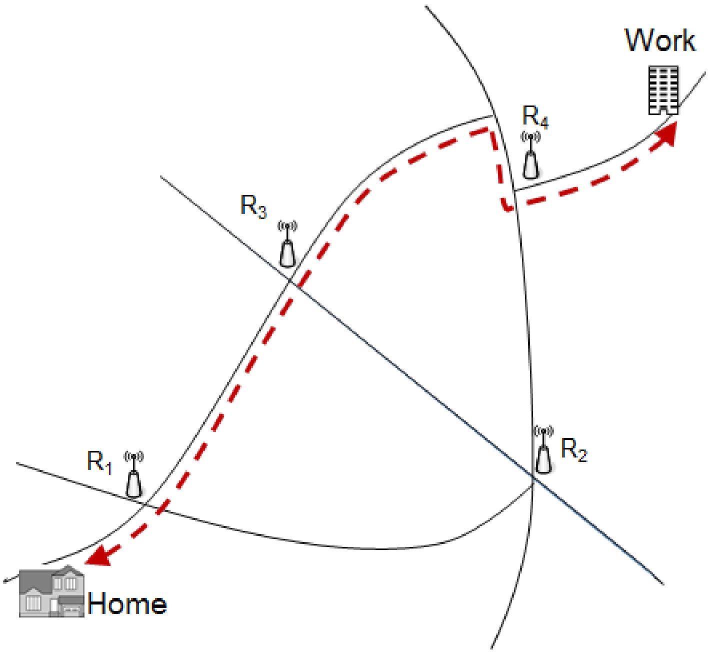


Fig. 1. A vehicle has a constant daily commute route (the red dashed line) between home and work place (or schools, daycares, superstores, etc), passing through RSUs R_1, R_3, R_4. One of these three RSUs will be designated as Agent RSU responsible for managing the reputation of the vehicle.

In this paper, we propose a further-enhanced community-based reputation system. We formalized the community-based reputation system, and we provide enhanced reputation-update mechanisms with concrete algorithms for the implementation of the proposed model. In the initial model presented in [3], each vehicle's behaviors can be monitored by only the RSUs located along the vehicle's commute, whereby the RSUs issue and send reputation-update messages to the "Agent RSU" of the vehicle; however, this approach seems impractical and unrealistic.

Vehicles can generate traffic messages anywhere at any time, and every vehicle's behaviors can therefore be evaluated instantly by any of its neighboring vehicles as well as RSUs. In our new model, the neighboring nodes of a vehicle evaluate the behavior of the neighboring vehicle and then generate a local proof for the behavior in an aggregated way. Subsequently, the Agent RSU of the vehicle updates the reputation score of the vehicle based on the local proofs and issues an updated reputation certificate. Since the reputation of a vehicle is determined by local proofs that are generated by neighboring nodes while the vehicle is being driven, both the reliability of the local proof and its delivery should be guaranteed. Lastly, we propose practical and efficient local-proof generation and delivery mechanisms for the establishment of a secure community-based reputation system.

This paper is organized in the following manner: We mention a selection of related works in Section 2, and then describe our assumptions and definitions in Section 3; in Section 4, we provide the detailed mechanisms that are required to implement the proposed reputation system; in Section 5 and Section 6, we analyze the security and the simulated performance of the proposed scheme; and in Section 7, we conclude the paper.

II. RELATED WORK

Many reputation strategies [4] have been proposed over the last decade. Most of these proposed strategies involve the employment of peer-to-peer (P2P) networks to establish a trust relationship [5] and to choose reliable peer nodes [6]-[7]. Reputation is commonly calculated and maintained by all cooperative neighbors through the evaluation of the past behaviors of a node and by reporting the corresponding information [8]. In order to support trust relationship in a mobile ad hoc networks [9], a group based reputation mechanism [10] and a distributed reputation mechanism [11] have been also proposed. In addition to reputation evaluation, reputation schemes that can encourage the cooperation of nodes have been studied, as well. In [12], a hierarchical reputation scheme associated with a pricing model has been suggested, and [13] analyzes the performance of models providing incentives for more favorable cooperative nodes.

Recently, many VANET reputation systems have emerged. These schemes can be classified into the following two approaches: (1) event (or data)-based reputation system (2) node-based reputation system. The first approach directly evaluates the trustworthiness of an instant event message, whereby the reputation can be either calculated immediately by its neighboring vehicles or aggregated by intermediate nodes during the propagation of the message. Authors in [14] proposed a data-centric trust establishment model that evaluates the trustworthiness of messages with evidence given from other peers. In [15], that the global reputation of a traffic event message is calculated by neighboring nodes based on a fuzzy-logic based reputation model, and a role-based reputation evaluation strategy for each traffic event message has been proposed in [16]. In order to generate an aggregated reputation evaluation by intermediate nodes, the reputation system in [17] makes use of opinion piggyback to enable confident decision on event packets. Opinions of intermediate nodes are simply appended for packet forwarding. Authors in [18] proposed an improved packet acceptance decision mechanism using a concept of trust token. During message relaying, a message receiving node can judge the reliability of the message relaying node using with the trust token. In [19], an enhanced opinion aggregation and propagation mechanism has been proposed.

The second approach updates the reputation of a node by accumulating the evaluations for the past behaviors of an individual node [20][21][22], whereby distrustful nodes with low reputations can easily be isolated and prevented from disseminating malicious messages. When an event message is delivered, even if there are not enough neighboring vehicles to verify the validity of the given message, an event message that has been created by a highly reputed vehicle can be trusted [21][22]. In addition, reputation-assisted data forwarding [20] can guarantee more accurate data delivery. Since data accuracy only depends on the reputation of the source vehicle, false-positive errors may occur, resulting in the generation of incorrect information by a vehicle with a favorable reputation. In [23], an RSU assisted reputation system has been proposed. Each vehicle keeps individual knowledge base (IKB) that stores its past experiences, and RSUs collect IKBs of vehicles passing through them. And the RSUs calculate the reputation of vehicles providing warning messages based on the IKBs and publishes the reputation list of vehicles. The scheme propose in [23] is similar to our proposed system in the sense of that RSUs take part in evaluating the reputation of vehicle. The RSUs in [23], however, evaluate a temporary reputation of the vehicles passing through the RSU based on the vehicle’s past experiences, but the RSUs in our propose model continuously manage the reputation of registered vehicles based on the locally aggregated evidence about the vehicle’s behaviors.

III. SYSTEM ARCHITECTURE

In this section, we illustrate the VANET environment that is compatible with our vehicular reputation system. The VANET environment under our consideration consists of the following essential elements:

- **Smart vehicle:** Smart vehicles are equipped with a tamper-proof on-board unit (OBU) for networking and computing; additionally, a digital-map GPS system for location detection is included in its design. Every smart vehicle maintains two types of key pairs for secure traffic-message communications. One is a permanent private–public key pair that is created and stored in its OBU by the car manufacturer. The permanent key pair will be used for the sole purpose of registering the car with the transportation safety authority. The other key pair is a temporary private–public key pair. This type of key pair can be periodically refreshed and is mainly used for securing traffic-message communications.
- **Roadside Unit (RSU):** RSUs are equipped with a tamper-resistant device that consists of storage, computing, and a wireless networking unit. Every RSU is authenticated and managed by the relevant governmental authority, and every unit is allocated a private–public key pair and certificate. An RSU already installs the public keys of other RSUs, and every RSU periodically broadcasts its ID and public key as beacon messages. The main role of an RSU is the transmission of traffic messages that are received from neighboring vehicles; however, RSUs can also be associated with the updating of a vehicle’s temporary public-key pairs and the issuance of the reputation certificates of particular vehicles. Any RSU that participates in the management of the reputation certificates of vehicles is defined as an Agent RSU and is denoted as “A-RSU”; every RSU is designed to be an A-RSU.
- **Certificate Authority (CA) (or Transportation Safety Authority):** Every smart vehicle is registered with the Transportation Safety Authority when it is purchased and this registration is managed periodically. The Authority also manages the RSUs and issues secure key pairs and certificates to those RSUs.

In the VANET environment, vehicles obtain and share useful traffic information via the communication of numerous traffic messages. The security strength, network resources, channels, and transmission methods are assigned differently according to the type of the traffic message. We simply classify a vehicle’s traffic messages into the following three types:

- **Typical traffic message:** This type of message contains typical traffic information such as speed, direction, location, and timestamp, and is very frequently broadcasted. Once this message type is broadcasted, it will not be forwarded any further. It requires a very low level of security.
- **Event message:** This type of message is generated occasionally whenever a sudden traffic event happens such as a traffic accident, sudden traffic jam, or abnormal road condition. The accuracy of the information is very important and this type can be propagated in multi-hops.
- **Relay message:** This type of message is received from another vehicle and should be routed to the message destination. Vehicles just carry and forward relay messages.

Vehicles communicate a variety of traffic-message types with other vehicles or RSUs while they are being driven, but we cannot assume that every one of these communications is reliable; that is, a portion of the messages is reliable but the other messages are likely to be inaccurate. In this paper, we classify the communication behaviors of vehicles into the following three groups:

- **Cooperative behavior:** Represents all kinds of cooperative communications such as useful and reliable traffic messages, forwarded or relayed messages, and replies to the requests of other vehicles. Vehicles elevate their reputations through cooperative behaviors.
- **Selfish behavior:** Represents all kinds of passive communications between vehicles; that is, a vehicle only listens to the traffic messages from a network, but does not participate in any cooperative communications. The selfish behaving vehicle never uses its resources for cooperative forwarding or the provision of useful traffic information for the benefit of others. We cannot say that such behaviors are “wrong,” but vehicles will not elevate their reputations through the enactment of selfish behaviors.
- **Malicious behavior:** Represents any attempts at making a VANET unavailable and insecure, including the fabrication of messages and the insertion of inaccurate information. A vehicle’s reputation will be degraded by malicious behaviors.

We are introducing a vehicular reputation system because the reputation of a source vehicle is a requisite for the efficient determination of whether a given traffic message is accepted or rejected, whereby the traffic messages that are provided by highly reputed vehicles are likely to be accepted. To ensure that traffic messages are accepted, vehicles should continuously accumulate reputation data regarding their behaviors; that is, an evaluation of an individual communication should be accomplished and the vehicle’s reputation should then be updated according to the evaluation.

The proposed community-based reputation system is an RSU-assisted vehicular reputation system. The key feature of our system is that the A-RSUs that are designated by vehicles are assigned the roles of managing vehicles’ reputation scores and issuing the corresponding reputation certificates that are based on the reputation scores of the vehicles. We define a reputation certificate in the following way:

[**Definition 1**] A **reputation certificate** is a public certificate that denotes a vehicle’s reputation score and is issued by the vehicle’s A-RSU. For vehicle *V*, the certificate contains *V*’s temporary public key, its reputation score, the issuance time, the expiration time, and the ID and signature of the A-RSU. Unlike other public certificates, the 24-hour validity period of a reputation certificate is relatively brief. *V*’s reputation certificate is updated periodically by *V*’s A-RSU, and *V* can obtain a new certificate whenever it passes by its A-RSU.

The reputation score of a vehicle is determined by the vehicle’s communication behaviors. Each communication behavior is instantly evaluated by neighboring vehicles or RSUs. Then, the neighbors generate a local proof about each behavior in an aggregated way and the proofs are delivered to the A-RSU of the vehicle. Every A-RSU collects the local proofs for its member vehicles over one day, and they then update the reputation scores of their member vehicles accordingly. Lastly, A-RSUs will issue new reputation certificates to reflect the updated reputation scores of its member vehicles. Consequently, the community-based reputation system requires detailed mechanisms for A-RSU designation, reputation-certificate generation and updates, and local-proof creation and delivery.

[**Definition 2**] The community-based reputation system is defined by the following four functions:

(1) Registration – Designation of an A-RSU

Vehicle *V* chooses its A-RSU from among the RSUs along its commute based on the unit that the vehicle passes by the most frequently or stays around for the longest time period. *V* then registers with the A-RSU with the help of a certificate authority. *V* obtains a secure token for the A-RSU designation from the certificate authority and shows the token to its A-RSU; if the token is valid, the A-RSU allows the enrollment of the vehicle for reputation management.

(2) Local-proof generation and delivery

Whenever *V* publishes event type of messages, the neighbors of *V* that detected such behaviors create a local proof for the behavior. The local proof is a sort of an evaluation certificate about *V*'s behavior. It contains an aggregated evaluation opinion of neighbors and it is digitally signed by the local proof creator. For computational and communicational efficiency, individual evaluations of neighbors are gathered by the most highly reputed neighbor, which then creates a local proof for *V*'s behaviors. The local proof will be delivered to *V*'s A-RSU through our proposed routing strategies.

(3) Reputation-certificate update

Each A-RSU updates the reputation scores of its member vehicles based on local proofs. An A-RSU first verifies the validity of a local proof upon receipt, and if the local proof is valid, it then renews the reputation score of the vehicle accordingly. The reputation score of a vehicle is updated whenever a local proof is given for a one-day period. At the (systemically predefined) certificate-generation time, the A-RSU will update the reputation certificates of every member vehicle based on the new reputation scores.

(4) Use of reputation certificate

Whenever *V* passes by its A-RSU, *V* obtains a new reputation certificate. During the validity period of the reputation certificate, *V* attaches its reputation certificate to its traffic messages. Any other vehicle can verify the reputation certificate by using the A-RSU's public key.

To make our reputation system reliable and practical, vehicles must not be able to fabricate their reputations; that is, it should be impossible to fabricate both a valid local proof and a valid reputation certificate. Reputation certificates must only be created by an authorized A-RSU, and a local proof must only be generated by a verifiable vehicle that has a valid reputation certificate. Our system is designed to satisfy the following security requirements:

- **Unforgeability:** A reputation certificate contains the signature of the A-RSU that issued the reputation certificate. The signature of the certificate must be valid for the reputation certificate to be considered valid. Unforgeability of the reputation certificate means that it is infeasible to forge a valid reputation certificate without knowing the private key of the A-RSU that is issuing the certificate. A local proof also includes the signature of the local-proof generator; therefore, it is infeasible to forge a valid local proof without knowing the private key of the vehicle issuing the local proof. Here, the vehicle must be in possession of a valid reputation certificate and the corresponding public key should be written in the certificate.
- **Verifiability:** Any entity in the VANET environment can verify the validities of reputation certificates and local proofs. A reputation certificate is verified with the public key of the A-RSU that issued the certificate. Any local proof is verified with the public key that is written in the reputation certificate of the local-proof generator.

IV. Community-based Reputation System

As is mentioned in Section 3, vehicles can broadcast any communication behaviors while they are being driven. We therefore explain the concrete protocols for carrying out each of the functions that is required for our community-based reputation system in this section.

A. Traffic Messages and Neighbor Group

As is described in Section 3, vehicles broadcast or forward traffic messages periodically or occasionally according to the type of traffic message. For the source vehicle V_S, the following basic traffic-message format is given:

$$TM_S = \{MType||[MID]||Timestamp||[GPS]||Information\}$$

where the items embraced with [] are optional, as these items can be selectively omitted in a message according to the message type. *MType* shows either "typical," "relay," or "event." *MID* is a randomly generated number and is assigned only for the "event" message type (it is omitted for typical or relaying messages). *Timestamp* is the traffic-message-generation time, *GPS* is the corresponding GPS information, and *Information* contains a description of the traffic situation at the time. In the case of a typical message, information such as speed and moving direction is included. In the case of a relay message, the relayed message will be included. In the case of an event message, the description of a particular traffic event such as a traffic jam, accident, or sudden brake will be included in addition to typical information. Lastly, V_S attaches its signature and reputation certificate to TM_S, as follows:

$$Final\ Data = \{TM_S||SIG(K_S^-, TM_S)||RCert_S\}$$

The broadcasting of typical messages does not affect the reputation of a vehicle. Alternatively, the relaying of messages is necessary for cooperative forwarding, and the publishing of a traffic event can be helpful for the safe driving of other vehicles. Local proofs are therefore only required for "relay" and "event" message types.

We assume that every vehicle maintains a neighbor list while it is being driven. For the source vehicle V_S, all of the vehicles belonging within the transmission range of V_S are set as V_S's neighbor group. The neighbor list basically contains information about a temporary public key and the reputation score and GPS of each neighboring vehicle. A vehicle can obtain such information from the typical messages that are broadcast periodically. The neighbor list is continuously updated whenever a typical message is received.

The following table summarizes all of the notations/acronyms and their descriptions that are used in this paper.

TABLE I.
Notations and Acronyms

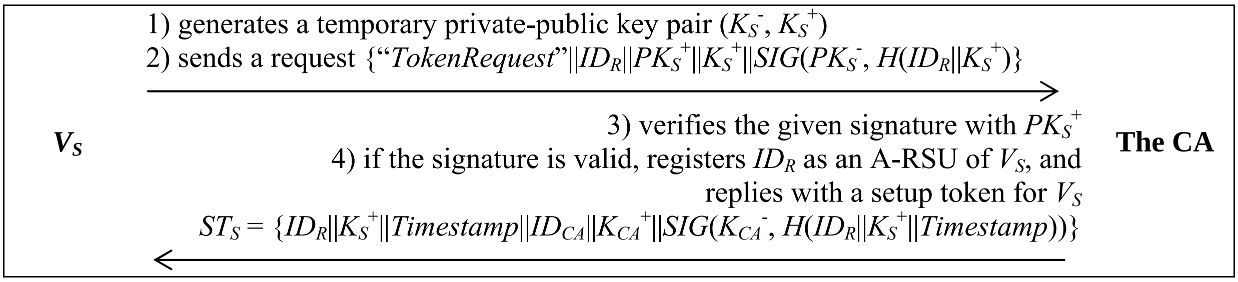
Notation	Description
(PK_i^-, PK_i^+)	A permanent private-public key pair of a vehicle V_i
(K_i^-, K_i^+)	A current temporary private-public key pair of a vehicle V_i
$SIG(K_i^-, M)$	A digital signature for a message M with a private key K_i^-
$H(\cdot)$	A cryptographic one-way hash function
$PKE(K_i^+, M)$	A Public key encryption for a message M with a public key K_i^+
LP_i	A local proof for V_i
$RCert_i$	A current reputation certificate of V_i
NL_i	A neighbor list of a vehicle V_i
N_C	A local proof creator
RSU	A roadside unit
A-RSU	An agent RSU that manages the reputation certificate of a vehicle

B. Secure Registration between a Vehicle and its A-RSU

To join our reputation system, the vehicle V_S should designate its A-RSU first. The A-RSU can be any one of the RSUs that are located along V_S's commute, but only one RSU can be registered as V_S's A-RSU so that V_S cannot collect multiple reputation certificates to protect against the unexpected possibility of a Sybil attack. In addition, the designation of a single A-RSU can make reputation-management simple and consistent. The RSU that V_S passes by the most frequently and stays around longer is ideally chosen as its A-RSU.

Once V_S chooses its A-RSU, V_S should perform a security association with the A-RSU to prove its legality and to protect against a duplicate registration. V_S must also obtain a setup token for the chosen A-RSU from the certificate authority, which prevents it from registering with multiple A-RSUs. The token-generation setup process can be carried out either online or offline.

Let the *ID* of the Agent RSU be ID_R. First of all, V_S identifies itself to the certificate authority using its permanent key pair. Since we assume that each vehicle's public key has already been enrolled along with the corresponding vehicle information to the certificate authority, V_S can prove its identification just by showing that it possesses a valid private key that corresponds to the registered public key. If V_S is a legal vehicle, then V_S performs the following token-generation setup protocol:



V_S

1) generates a temporary private-public key pair (K_S^-, K_S^+)
2) sends a request $\{\text{“}TokenRequest\text{”}||ID_R||PK_S^+||K_S^+||SIG(PK_S^-, H(ID_R||K_S^+))\}$

3) verifies the given signature with PK_S^+
4) if the signature is valid, registers ID_R as an A-RSU of V_S, and replies with a setup token for V_S

$$ST_S = \{ID_R||K_S^+||Timestamp||ID_{CA}||K_{CA}^+||SIG(K_{CA}^-, H(ID_R||K_S^+||Timestamp))\}$$

The CA

When V_S passes by its A-RSU, it sends a setup request that is signed by V_S's temporary private key along with the setup token, as follows:

$$Setup\ Request = \{\text{“}SetupRequest\text{”}||Timestamp||ST_S||SIG(K_S^-, H(ID_R||K_S^+||Timestamp))\}$$

The A-RSU first verifies the validity of the setup token with $K_{CA}{}^{+}$, followed by the verification of the signature validity upon request with the public key $K_S{}^{+}$ that is specified in the token. If both signatures are valid, an initial reputation certificate is generated for V_S. This setup step occurs only once after the setup request is given to an A-RSU.

C. Local-Proof Generation

Once registration has been completed, the A-RSU collects and manages V_S's communication behaviors. V_S's activities should be detected by other vehicles or RSUs, and the detected information should be delivered to the A-RSU as well. The use of "local proof" regarding V_S's behavior is therefore inevitable, and V_S's neighboring vehicles should be able to generate the proof. Notably, V_S's behavior is useless if there is no neighboring smart vehicle; therefore, we suppose that V_S will only generate event or relay messages in the presence of neighboring vehicles.

Local-proof generation and delivery are the most essential and the most challenging parts of our scheme. To make this process practical and reliable, we should be able to answer the following three questions:

(1) How can neighbors generate a local proof?
(2) How can the reliability of a local proof be ensured?
(3) How can the delivery of a local proof to the corresponding A-RSU be guaranteed?

To solve the first two questions, we propose the allowance of highly reputed neighbor to issue an aggregated local proof based on the agreement of neighbors. To address the third issue, we propose two types of local-proof delivery mechanisms that can minimize communication-traffic overheads while also guaranteeing delivery. In this section, we propose two protocols for the following: (1) creating the local proof, and (2) efficient local-proof delivery.

The reputation score of a vehicle (denoted as V_S) will mainly be decided according to the local proofs that are generated by V_S's neighbor nodes. In the case where V_S just relays a message, an acknowledgement of the relayed message from a receiver is enough for the local proof. In the case where V_S provides newly detected traffic information, a simple agreement from the neighbors regarding the reliability of the information is necessary. We therefore introduce an aggregated local-proof-generation mechanism for V_S's event messages in this section.

1) Selecting a local-proof creator

First, if there is a single neighboring vehicle around V_S, then the neighbor undoubtedly becomes a local proof creator whereby it will create the local proof for V_S's event message. Then, V_S will reply to the neighbor with an acknowledgement message as well. The acknowledgement message will also be used by the neighbor's A-RSU to increase the reputation of the neighbor (local-proof creator) in the future. The neighbors will therefore actively take part in the creation of a local proof because their reputation can be increased. V_S cannot increase its reputation with the local proof because the reliability of the local proof is still suspicious as it contains only one opinion; therefore, it will not be used to update V_S's reputation score.

Secondly, with a minimum of two neighboring vehicles, an aggregated local proof can be generated. Our simple strategy is to elect a local proof creator among the neighbors so that the local proof creator generates a local proof based on the agreement of the two vehicles with the highest reputation scores in the neighbor group. Since the local proof creator should collect opinions of neighbors, the qualification of the local proof creator should satisfy at least the following three conditions: (1) it must be highly reputed, (2) it is moving the same direction of V_S, and (3) it moves with the similar speed to the average speed of neighbors.

The biggest issue here is whether it is possible to find the local proof creator in the neighbor group without transmitting additional messages among the group. As we previously mentioned, V_S is supposed to keep a neighbor list that contains the reputation scores, temporary public keys, moving direction, and GPS locations of neighboring vehicles. Among the neighbors, some of the vehicles that are located within the half-transmission range of $\mathrm{V_S}$ can be denoted as core neighbors, as illustrated in Figure 2.

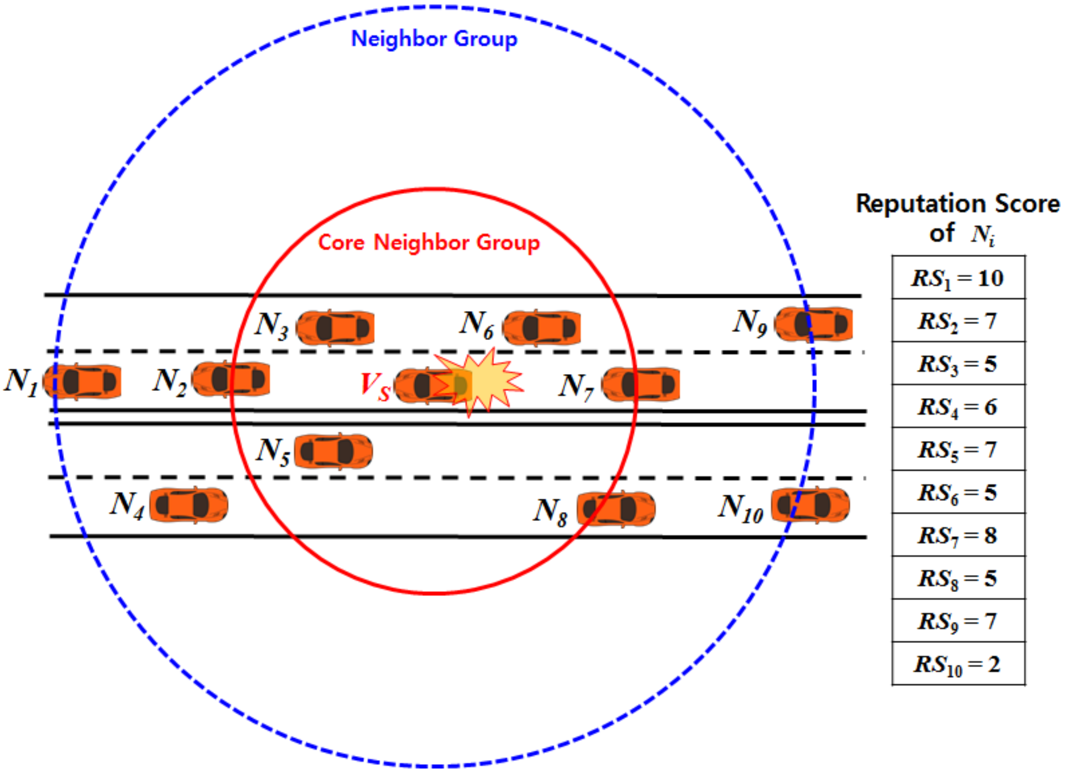


Fig. 1. Neighbor group of a vehicle V_S. V_S generates an event message about its flat tire. N_2, N_3, N_5, N_6, N_7, and N_8 are core neighbors of V_S, and they are all included in each other's neighbor lists; therefore, N_7 becomes a local proof creator. N_1 does not become a local-proof creator because it is not a core neighbor of V_S, even though its reputation is the highest. N_7 can receive local proofs from N_2, N_3, N_5, N_6, N_8, N_9, and N_{10}.

These core neighbors have two useful features that mean that these neighbors can receive V_S's message more quickly than those neighbors that are located further away, and all of these core neighbors are included in each other's neighbor lists. Based on this fact, the most highly reputed vehicles can be automatically identified just by consulting each vehicle's neighbor list. On top of it, a vehicle can calculate a relative velocity with its neighboring vehicles based on speeds and moving directions of vehicles. Let speeds of two vehicles V_i and V_j be S_i and S_j, respectively. The relative velocity $RV_{i,j}$ is calculated by the following equation:

$$RV_{i,j} = \sqrt{s_i^2 + s_j^2 - 2 \cdot s_i \cdot s_j \cdot \cos\varphi} \tag{1}$$

Here, φ means the direction difference between two vehicles. If φ is 0 since two vehicles are heading for the same direction, $RV_{i,j}$ is the smallest value as we expected. However, if φ is increased according to the different direction of the two vehicles, $RV_{i,j}$ is gradually increased. From the equation (1), $RV_{i,j}$ converges to 0, as two vehicles move in the same direction at the similar speed. Consequently, a local proof creator is determined by the following algorithm.

Let the neighbor list of V_S be NL_S. Each vehicle in NL_S is denoted as N_i for $1 \leq i \leq n$, and N_i's neighbor list can be denoted as NL_i. For each N_i, if N_i satisfies all of the following four conditions when it received V_S's event message, then it announces itself as a local-proof creator to its neighbors; we denote a local-proof creator as N_C in this paper:

(1) It is a core neighbor; and
(2) $RV_{i,S} <$ α, where α can be systemically predefined; and
(3) Its reputation score is the highest among neighbors satisfying the above two conditions in NL_i; and
(4) It is the most closely located to V_S if there are plural neighbors having the same reputation score.

Every $N_i \in NL_S$ can determine whether or not it is qualified to become a local-proof creator simply by consulting its neighbor list. When N_i received N_C's announcement, N_i can also verify whether or not N_C satisfies at least the first three conditions. The fourth condition is required to distinguish one local-proof creator among the most highly reputed neighbors. Once N_C has been announced in the neighbor group, each N_i sends its ACK message to N_C where the ACK contains N_i's opinion about the V_S's behavior. .

We may consider another method for the selection of a local-proof creator, whereby V_S designates and announces the most highly reputed neighbor in NL_S. This approach seems more efficient because neighbors do not need to find a local-proof creator in their neighbor lists. The problem, however, is that V_S could be dishonest. V_S may be able to designate any compromised neighbor as a local-proof creator for the purpose of increasing V_S's reputation. Our proposed strategy is advantageous because the local-proof creator is chosen by a reputation comparison that is independent of the process performed by distributed neighbors. Our strategy accompanies an implicit agreement by the current neighbors without requiring additional message transmissions.

2) Local-proof generation

Once N_C has been decided, N_C generates an aggregated local proof with N_i. Suppose that the following event message from V_S is given:

$Event\ Message = \{TM_S||SIG(K_S^-, TM_S)||RCert_S\}$ where $TM_S = \{MType||MID||Timestamp||GPS||Information\}$

First, every N_i creates its own ACK message about the behavior of V_S, and sends the ACK message to N_C. The ACK message includes N_i's opinion regarding whether the given *Information* is "good" or "bad." If the given information is useful and agreeable, then the opinion is "good", otherwise, the opinion sets as "bad." N_i sends the ACK message encrypted with N_C's public key, as follows:

1. N_i	1) Every N_i in NL_S creates its own ACK message and signature as follows: $ACK_i = \{K_S^+||MID||OP||Timestamp\}$ and $SIG_i = SIG(K_i^-, H(ACK_i))$ 2) Every N_i except N_C sends its ACK message to N_C as an encrypted way: $\{PKE(K_C^+, ACK_i)||SIG_i||RCert_i\}$ $\longrightarrow$	**N_C**

N_i can obtain K_S^+ from the reputation certificate of V_S, and the *MID* is given in the event message of V_S. *Timestamp* is the generation time of the ACK message. *OP* is the opinion of N_i. Since ACK_i is encrypted with the public key of N_C, only N_C can reveal the ACK message.

N_C chooses only one of the neighbor ACK messages that are issued based on the reputation that is the highest among all of the ACK messages that are received. Finally, N_C generates an aggregated local proof as follows:

2. N_C	1) N_C chooses only one ACK message, denoted as ACK_1, having the highest reputation. 2) N_C decrypts ACK_1 with its private key, and verifies the validity of SIG_1 with the public key given in N_1's reputation certificate. 3) N_C generates and sends a local proof for V_S such as $LP_S = \{ ACK_1 \,||\, ACK_C \,||\, SIG_1 \,||\, SIG_C || \, RCert_1 || \, RCert_C\}$ $\longrightarrow$	**V_S**

Consequently, the local proof consists of two distinct opinions of the most highly reputed vehicles in the neighbor group. The validity of LP_S can be proven by verifying two signatures of SIG_1 and SIG_C using with the public keys given in the reputation certificates of N_1 and N_C. Then, V_S replies its acknowledgement to N_C and N_1 as follows:

V_S	1) create an acknowledge message $ACK_S = \{K_C^+||K_1^+||LP\text{-}ACK||Timestamp\}$ 2) generate a signature $SIG_S = SIG(K_S^-, H(ACK_S))$ 3) Send $\{ACK_S \,||\, SIG_S \,||\, RCert_S\}$ $\longrightarrow$	**N_1, N_C**

V_S's acknowledgements are then used at a later time by the corresponding A-RSUs to update the reputation scores of N_1 and N_C; also, V_S's acknowledgement is a local proof of the cooperative actions of N_1 and N_C. N_C can send back its acknowledgements to the other neighbors who sent ACK messages to N_C, and N_1 and N_C can keep the message and deliver it to their own A-RSUs.

D. Delivery of Local Proofs

The next consideration is the delivery of local proofs. It should be possible to deliver a local proof to its corresponding A-RSU without any loss of delivery within a reasonable time frame. Since we assume that our reputation certificates are refreshed every 24 hours, it seems reasonable to presume that local proofs would reach the corresponding A-RSUs before the next reputation-certificate update. We therefore allow, at most, up to a 24-hour delay of the local-proof delivery based on the local-proof generation time. An efficient routing method whereby delays are tolerable but delivery is guaranteed and communication overheads are minimized needs to be devised. In this section, we propose an efficient local-proof-delivery mechanism that is especially suitable for our reputation system.

1) Self-delivery of the local proofs

For the enactment of our simple strategy, the source vehicle carries and delivers all of the collected local proofs to its A-RSU; if it is necessary, and only then, the local proofs can be routed to the A-RSU by arbitrary intermediate vehicles. Since the A-RSU is located along the commute of the source vehicle, all of the proofs regarding the vehicle will be delivered to the A-RSU whenever the vehicle passes by the unit. Additionally, no further communication overheads for routing are incurred with such a self-delivery mechanism, and the self-delivery of local proofs means that both delivery accuracy and communication efficiency can be ensured at the same time. The only problem is that the source vehicle can prevent the delivery of some local proofs to its A-RSU, negatively affecting its reputation score; in such a case, the reputation system cannot work correctly. To prohibit any manufacturing of reputation scores, our proposed routing mechanism will be used.

As described in the previous section, each local proof contains two opinions that are evaluated by two distinct neighbors for an individual event of V_S. If all of the contents of "*OP*" in a local proof are "good", then the source vehicle will willingly deliver the proof to its A-RSU to increase its reputation score. Additional routing for the delivery of the local proof is unnecessary. It is still suspicious if one is "good" and the other is "bad," so there is no effect on the decision process for the reputation score; therefore, this type of local proof is not necessary for the inevitable delivery to the A-RSU. If all of the contents of the "*OP*" are "bad," then the reputation score will definitely be negatively affected, so the vehicle may deliberately avoid delivering the local proof to its A-RSU; in this case, the local proof should be routed to the corresponding A-RSU by the neighboring vehicles. The local-proof creator begins routing the local proof if all of the contents of the local proof are "bad."

2) The proposed routing strategy

For our basic routing policy, the highly reputed vehicle carries and forwards the local proof. The local-proof creator initially carries and propagates the local proof to its neighbors. Then, one of the neighbors, which is the most highly reputed vehicle, will propagate the local proof again. In this way, the local proof can be routed to its corresponding A-RSU by highly reputed intermediate vehicles. Since vehicles of a high reputation will be more likely to take part in cooperative routing, the local proof will be delivered to the A-RSU with a relatively higher probability.

The delivery of a local proof, however, cannot be guaranteed without a consideration of network-situation factors such as channel conditions, traffic density, relative velocity and direction of the vehicles. Even when the reputation of a neighboring vehicle is fairly high, the local proof will not be given to the highly reputed neighbor unless the network-channel condition, traffic density, relative velocity and direction of the vehicles support the delivery. Direction of movement and speed of a vehicle are very essential parameters for the calculation of the routing strategy in case of VANETs. The relative velocity between vehicles is inversely proportional to the expected link duration of the route. Consequently, our delivery mechanism chooses the next intermediate vehicle among the neighbors based on the reputation and the channel condition as well as the vehicles correlation (i.e., traffic density, relative velocity and direction of the vehicles). Let the current *LP* carrier be V_i, and let the neighbors of V_i be $NL_i = \{N_1, N_2, \ldots, N_m\}$. For $j=\{1,\ldots,m\}$, C_{ij} is the network-channel-condition value between V_i and N_j, and RS_j is the reputation score of N_j. $RV_{i,j}$ is the relative velocity between V_i and N_j. The next intermediate vehicle for the *LP* propagation can be chosen by the following delivery function $D_i(NL_i)$:

$$D_i(NL_i) = \max_{j=1,\ldots,m}\{\alpha \cdot c_{i,j} + (1 - \alpha) \cdot RS_j\}/RV_{i,j} \mid \delta(i) > \theta, \tag{2}$$

where α is a systemically predefined weight and $\delta(i)$ is the density estimation value. $\delta(i)$ is defined as follows:

$$\delta(i) = \frac{\sum_{j\in NL_i}\sum_{t_i\in[t,t+T)} ETX_{i,j}}{\sum_{l\in L}\omega_l \cdot A_l(j)},\ \forall i, \forall j, \forall l \tag{3}$$

We sample the vehicle density at random intervals with the uniformly distributed duration. At each epoch t_i, the vehicle V_i measures expected transmission count (ETX) with the center position and the radius of the target area, $A_l(j)$. To avoid flooding at the receiver vehicles, the reply message (i.e., AcK) time is randomized. V_i utilizes the number of AcKs and its own position at each epoch t_i. ω_l is the weight according to the target area and proportional with the distance between V_i and N_j. Lastly, the neighbor N_j that maximizes $D_i(NL_i)$ becomes the next intermediate node.

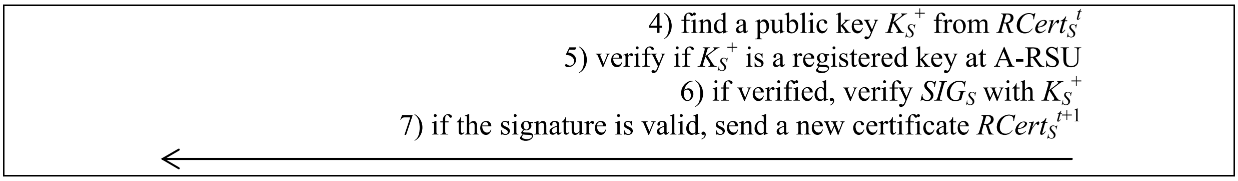
E. Reputation-Certificate Update

The main task of an A-RSU is to issue and update the reputation certificates for its member vehicles. The A-RSU constantly collects local proofs and repeats the issuance of reputation certificates periodically. The reputation certificates are supposed to be updated every 24 hours, so the A-RSU performs two types of tasks. First, the A-RSU refreshes the reputation score of a vehicle instantly whenever a local proof is delivered to it. Secondly, the A-RSU generates reputation certificates, in batches, for its member vehicles based on each vehicle's up-to-date reputation score at the end of each update period (for example, at every midnight). We assume that individual local proofs are stored at the A-RSU for a maximum time period of 24 hours based on the local-proof-generation time, and that they will be eliminated thereafter for storage efficiency. The following detailed algorithm is proposed for the updating of a new reputation certificate:

```
Let $\boldsymbol{V_j}$ be a member vehicle of an A-RSU for $j$={1,…,$n$};
Let $\boldsymbol{LP_j}$ be the local proof for $V_j$;
Let $\boldsymbol{RS_j}$ be the current reputation score of $V_j$;
Let $\boldsymbol{RCert_j^i}$ be the $i$-th reputation certificate of $V_j$;
while (a time interval $\boldsymbol{i}$) {
  if ($LP_j$ is given && $LP_j$ is alive) {
    reveal "Message-ID" of $LP_j$;
    if ( $LP_j$ of the same "Message-ID" exists already)
      eliminate $LP_j$;
    else {
      verify two signatures used in $LP_j$;
      if (both signature are valid) {
        reveal "Behavior-Type" in $LP_j$;
        if (both "Behavior-Type" are good) $RS_j$++;
        if (both "Behavior-Type" are bad) $RS_j$--;
        save $LP_j$ in the storage;
      }
    }
  }
  else
    eliminate $LP_j$;
  if (Reputation certificate generation time) {
    generate new certificates $\boldsymbol{RCert_j^{i+1}}$ for $\forall V_j$ based on $RS_j$;
    i++;
  }
}
```

Vehicles can obtain their new certificates whenever they pass through their A-RSUs. Since we assumed that the A-RSUs are located on the commute routes of vehicles, vehicles can periodically obtain their new certificates. A new certificate update request protocol to obtain a new reputation certificate $RCert_S^{t+1}$ that will be used for the (t+1)th time period with the tth reputation certificate $RCert_S^t$ is as follows:

$\boldsymbol{V_S}$	1) create a certificate update message URC_S = {"Certificate Update" ‖*Timestamp*} 2) generate a signature $SIG_S = SIG(K_S^-, H(URC_S))$ 3) send { URC_S ‖ SIG_S ‖ $RCert_S^t$} $\longrightarrow$	**A-RSU**



4) find a public key K_S^+ from $RCert_S^t$

5) verify if K_S^+ is a registered key at A-RSU

6) if verified, verify SIG_S with K_S^+

7) if the signature is valid, send a new certificate $RCert_S^{t+1}$

V_S signs the certificate update message with its temporary private key to prove the V_S is the owner of the current reputation certificate. If the public key given in the reputation certificate is the public key of a legal vehicle registered to A-RSU, and if the given signature is valid, too, then A-RSU sends a new certificate for next time period to V_S. Thus, only a legal vehicle who has been already registered to A-RSU and possesses a valid reputation certificate can obtain updated reputation certificate.

In addition, vehicles can refresh their temporary public key pairs while updating their reputation certificates. V_S generates a new temporary public key pair, which is denoted as ($K_S^{+'}$, $K_S^{-'}$). Then, V_S creates a modified update request message that includes a new temporary public key as follows:

$$URCK_S = \{\text{"Certificate Update with New Key"} \,||\, K_S^{+'} \,||\, TimeStamp\}$$

And V_S signs $URCK_S$ with its current private key K_S^-. The following steps are the same. If the request message is valid, then the new key $K_S^{+'}$ will be used in creating the (t+2)th reputation certificate. V_S will use its new public key pair from the (t+2)th time period.

V. Security analysis

We analyze the security of our proposed reputation system in four aspects of unforgeability, verifiability and privacy.

A. Unforgeability

The reliability of our proposed reputation system depends on the reliability of local proofs. That is, the local proof has not to be fabricated nor forged arbitrarily. Our system lets only the local proof creator generate a local proof, and in addition, the local proof aggregates two opinions created by two distinct highly reputed vehicles. Each opinion is digitally signed and the reputation certificate of the opinion generator is also included in the local proof. Since the reputation certificate contains the signature of the vehicle's A-RSU, it is eventually infeasible to forge or fabricate a valid reputation certificate without knowing the private key of the A-RSU. Subsequently, it is infeasible to fabricate a valid local proof without knowing the private key paired with the public key given in a valid reputation certificate. Thus our reputation system is secure since only legal vehicles can join the reputation system with valid reputation certificates.

B. Verifiability

Once a local proof has been reached to an A-RSU, the A-RSU can determine the validity of the local proof by verifying validity of the signatures in the local proof. Each RSU is supposed to know the public keys of other RSUs, so, the RSU can verify the validity of the reputation certificate generated by any RSU. If the signature of the local proof can be verified by the public key written in the reputation certificate, then RSU can decide as the local proof is valid. The RSU cannot know exactly the identification of the vehicle signed to the local proof though, but it can know at least that the local proof is created by a legal vehicle possessing a valid reputation certificate.

C. Privacy

On top of the two basic security requirements, we show that vehicle's privacy can be still preserved in spite of using the reputation certificate. Each reputation certificate contains each vehicle's temporary public key but the public key is randomly chosen public key regardless of the real identification of the vehicle. Thus, vehicles cannot be identified from the reputation certificates. Since every traffic message includes each vehicle's reputation certificate, it may be regarded as the driving trajectories of vehicles would be traced, so, vehicle's privacy can be invaded. If vehicles use fixed temporary public keys continuously in their reputation certificate update, such a problem can be issued. Vehicles, however, refresh daily their reputation certificates, and they can also renew their temporary public keys at every certificate update moment. Thus, the privacy about vehicles and their trajectories can be partially preserved.

VI. EXPERIMENTS

A. *Mobility Models*

In the proposed scheme, each vehicle utilizes the reliable route from source vehicle to destination vehicle based on DSR from RSU recommendation and the proposed route strategy. Using eq. (2), the proposed scheme can select the best link duration for spatial dependent of movement among vehicles and temporal dependence of movement of a vehicle over time. Plus, to reflect on existence of barriers or obstacles constraining mobility, we use a realistic mobility model (i.e., Reference Point Group Mobility (RPGM) [24]). Among many routing protocols, the reason why we select DSR in RSU is as follows. The authors of [24] concluded that on-demand protocols (i.e., DSR and AODV) performed better than table driven ones (i.e., DSDV) at high mobility rates such as VANETs. The authors of [25] additionally consider the traffic demand as well as mobility degree between on-demand protocols and table driven ones. They found DSR is more efficient inspects of small overhead, low traffic load and medium mobility rates. Therefore, we focus on DSR to verify our proposed scheme based on Reference Point Group Mobility (RPGM) [24]. In RPGM, the rate of link changes was used to express a few group mobility patterns as well as Random Waypoint. However, the frequency of the rate of link changes in RPGM is smaller than only Random Way. That is, based on the characteristics of RPGM mobility model, we can focus on spatial dependence, geographic restrictions and temporal dependence.

B. *Protocol Performance Metrics*

To evaluate the effect of mobility model with the proposed scheme, we evaluate extensive simulations with the network simulator (ns-3). Experimental results show that our proposed scheme outperforms other solutions in terms of link duration, throughput (ratio of the number of packets delivered to the number of packets sent) and routing overhead (number of routing control packets sent). System parameters for the simulation are like followings. The transmission range of the nodes was 250 m. The traffic is 20 CBR sources and 30 connections. The source–destination pairs were chosen randomly and we utilized different random seeds to generate three different traffic patterns with the same number of sources and connections. The data rate was 4packets/s. And, the packet size was 64 bytes.

As shown in Figure 3, the proposed scheme with the enhanced RPGM mobility model has a higher value than other schemes. Specially, for the Freeway and Manhattan value is similar to Random Waypoint in maximum speed of vehicles because of the opposite direction of motion and high relative speeds. Therefore, these are not realistic and inefficient in VANET. In Figure 4 and 5, the proposed scheme shows a difference of almost 40% in throughput and routing overhead from other schemes, respectively. Since the proposed scheme selects the best link with eq. (2) as well as an efficient DSR on top of RSU under realistic mobility model. That is, the proposed scheme keeps long link duration based on optimal local decision on top of the rough centralized DSR route from RSU.

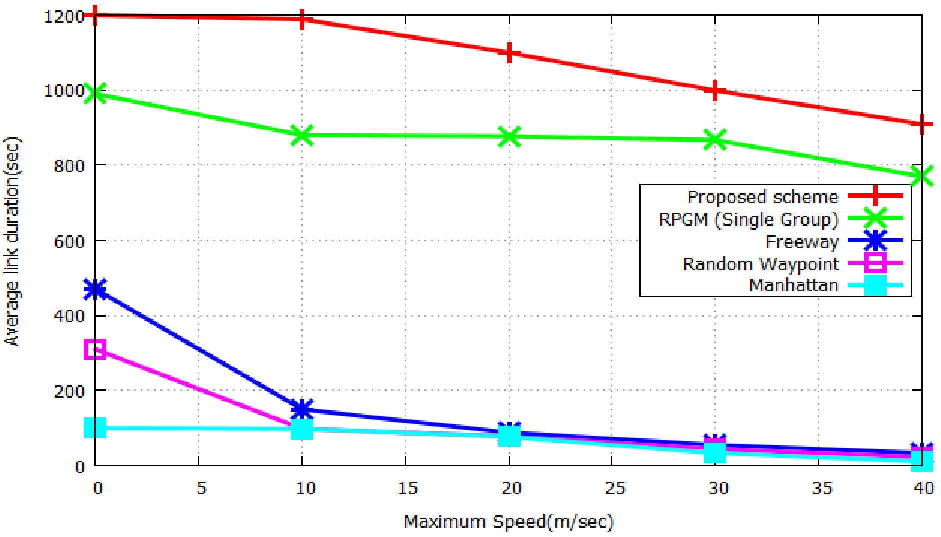


Fig. 3. Average link duration

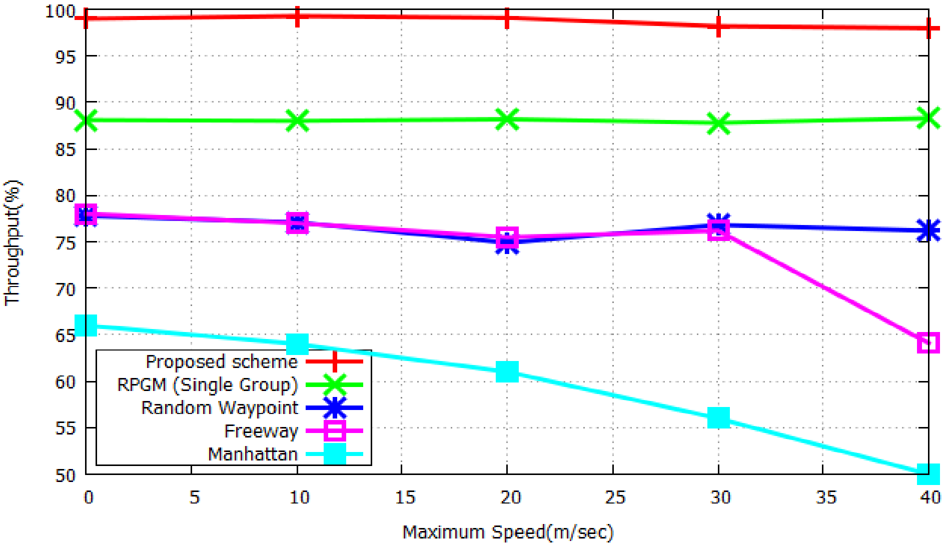


Fig. 4. Throughput

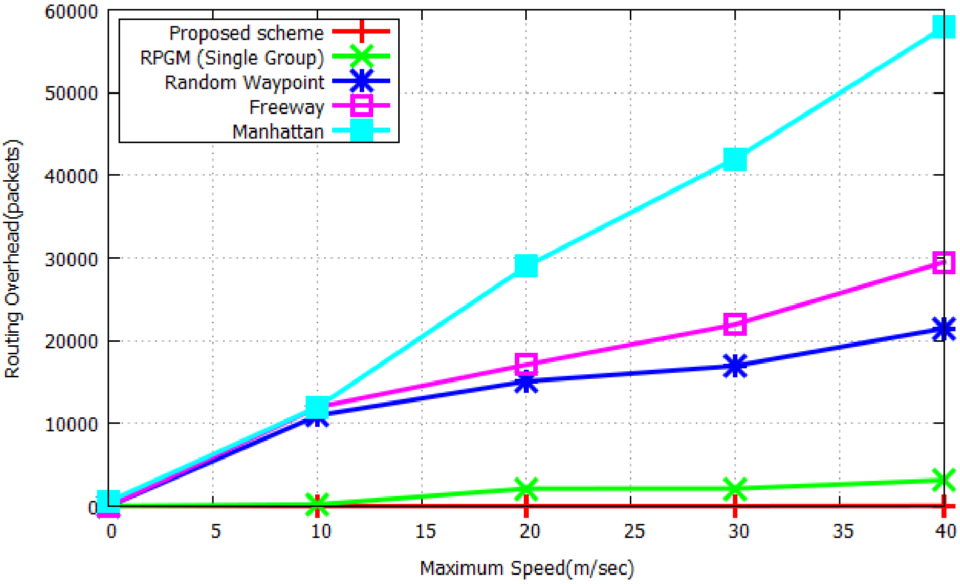


Fig. 5. Routing overhead

VII. Conclusion

In the proposed vehicular reputation system, any vehicle's reputation can be evaluated by neighboring vehicles in real-time. An aggregated local proof can be obtained from neighboring vehicles based on the agreement of the neighbors. The local proofs are collected at the vehicle's A-RSU, and the A-RSU periodically issues reputation certificates based on the local proofs. The local proofs should be delivered to the vehicle's A-RSU as fast as possible without loss of delivery. In order to figure it out, we have also proposed an efficient local proof routing strategy which can maximize the delivery throughput. We have allowed that vehicles carry their local proofs by themselves in order to reduce the communication overheads for delivering the local proofs. But negative local proofs should be still routed to the A-RSU by other vehicles. Since both the reputation of vehicle and the network channel conditions are considered for choosing next intermediate vehicle in the local proof routing, our simulated results show that out proposed mechanism has the best throughput.

We need further research about a way of using instant local proofs evaluated by neighboring vehicles securely in real time before those local proofs are reflected into the reputation certificate. Even in our current model, local proofs can be used instantly together with traffic messages, but the size of traffic message increases as local proofs are accumulated, and the verification costs of the local proofs increases proportionally. Thus, we will keep researching to find an efficient way for reflecting the local proofs instantly on the reputation certificate while minimizing the verification costs.

ACKNOWLEDGMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No.NRF-2014R1A1A3053491) and (NRF-2014R1A1A1003562).

REFERENCES

[1] Y. Qian, N. Moayeri, "Design Secure and Application-Oriented VANETs," in *Proc. of IEEE VTC'2008-Spring,* 2008.

[2] M. Raya, J-P. Hubaux, "Securing vehicular ad hoc networks," Journal of Computer Security. *Special Issue on Security of Ad Hoc and Sensor Networks,* vol. 15, no. 1, pp. 39-68, 2007.

[3] S. Park, B. Aslam, C. C. Zou, "Long-term reputation system for vehicular networking based on vehicle's daily commute routine," in *Proc. of the IEEE Consumer Communications and Networking Conference (CCNC),* 2011, pp. 436-441.

[4] M. Ibrohimovna, S. H. Groot, "Reputation-based Systems within Computer Networks," in *Proc. of Int'l Conference on Internet and Web Applications and Services*, 2010.

[5] C. Tian, J. Cheng, "Building an Efficient Distributed Reputation Scheme for Peer-to-Peer Networks," in *Proc. of IEEE Int'l Symposium on Information Science and Engineering,* 2008, pp. 285-288.

[6] F. Cornelli, E. Damiani, D. C. di Vimercati, *et al.*, "Choosing Reputable Servants in a P2P Networks," in *Proc. of Int'l World Wide Web Conference,* 2002, pp. 441-449.

[7] E. Damiani, D. C. di Vimercati, S. Paraboschi, *et al.*, "A Reputation-based Approach for Choosing Reliable Resources in Peer-to-Peer Networks," in *Proc. of the 9th ACM conference on Computer and Communications Security,* 2002, pp. 207 – 216.

[8] S. Buchegger, J. Mundinger, J. L. Boudec, "Reputation Systems for Self-Organized Networks: Lessons Learned," *IEEE Technology & Society Magazine,* 2007.

[9] C. Zouridaki, B. L. Mark, R. K. Thomas, "Robust Cooperative Trust Establishment for MANETs," in *Proc. of ACM workshop on Security of ad hoc and sensor networks,* 2006, pp. 23–34.

[10] X. Wu, J. He, F. Xu, "A Group-based Reputation Mechanism for Mobile P2P Networks," LNCS 2005, vol. 3828, pp. 651-659.

[11] J. J. Jaramillo, R. Srikant, "DARWIN: Distributed and Adaptive Reputation Mechanism for Wireless Ad-Hoc Networks," in *Proc. of ACM Int'l Conference on Mobile Computing and Networking,* 2007, pp. 87-98.

[12] H. Shen, Z. Li, "ARM: An Account-based Hierarchical Reputation Management System for Wireless Ad Hoc Networks," in *Proc. of IEEE Int'l Conference on Distributed Computing Systems Workshops,* 2008.

[13] Z. Li, H. Shen, "Analysis the cooperation strategies in mobile ad hoc networks," in *Proc. of IEEE Int'l Conference on Mobile Ad Hoc and Sensor Systems,* 2008, pp. 880-885.

[14] M. Raya, P. Papadimitratos, V. D. Gligor, J-P. Hubaux, "On Data-Centric Trust Establishment in Ephemeral Ad Hoc Networks," in *Proc. of IEEE INFOCOM*, 2008.

[15] Q. Ding, X. Li, "Reputation Management in Vehicular Ad Hoc Networks," in *Proc. of Int'l Conference on Multimedia Technology (ICMT)*, 2010.

[16] Q. Ding, M. Jiang, X. Li, X. Zhou, "Reputation-based Trust Model in Vehicular Ad Hoc Networks," in *Proc. of Int'l Conference on Wireless Communications and Signal Processing (WCSP),* 2010.

[17] F. Dotzer, L. Fischer, P. Magiera, "VARS: A Vehicle Ad-Hoc Network Reputation System," in *Proc. of IEEE Int'l Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM)*, 2005.

[18] Z. Wang, C. Chigan, "Cooperation Enhancement for message transmission in VANETs," *Int'l Journal of Wireless Personal Communications,* 2007, vol. 43, no. 1, pp. 141 – 156.

[19] J. Zhang, C. Chen, R. Cohen, "A Scalable and Effective Trust-Based Framework for Vehicular Ad-Hoc Networks," *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications,* 2010, vol. 1, no. 4, pp. 3-15

[20] N. Li, S. K. Das, "RADON: Reputation-Assisted Data Forwarding in Opportunistic Networks," in *Proc. of the International Workshop on Mobile Opportunistic Networking (ACM/SIGMOBILE MobiOpp),* 2010.

[21] A. Patwardhan, A. Joshi, T. Finin, Y. Yesha, "A Data Intensive Reputation Management Scheme for Vehicular Ad Hoc Networks," in *Proc. of Int'l Conference on Mobile and Ubiquitous System,* 2006.

[22] R. K. Schmidt, T. Leinmuller, E. Schoch, A. Held, G. Schafer, "Vehicle Behavior Analysis to Enhance Security in VANETs," in *Proc. of IEEE Workshop on Vehicle to Vehicle Communications (V2VCOM),* 2008.

[23] C. P. Fernandes, I. Simas, E. R. Mello, M. S. Wangham, "RS4VALNETs – A Decentralized Reputation System for Assessing the Trustworthiness of Nodes in Vehicular Networks," in *Proc. of IEEE Int'l Wireless Communications and Mobile Computing Conference (IWCMC),* 2015.

[24] X. Hong, M. Gerla, G. Pei, C-C. Chiang, "A group mobility model for ad hoc wireless networks," in *Proc. of ACM/IEEE MSWiM,* 1999.

[25] J. Broch, D. A. Maltz, D. B. Johnson, Y. C. Hu, J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proc. of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking,* 1998.

[26] S. R. Das, C. E Perkins, E. M. Royer, "Performance comparison of two on-demand routing protocols for ad hoc networks," in *Proc. of INFOCOM,* 2000.

AUTHOR PROFILE

Hayoung Oh received the B.S. degree in Computer Science from Duksung Womans University and the M.S. degree in the School of Computer Science and Engineering from Ewha Womans University in 2002 and 2006 respectively. And she received the Ph.D. degree in Computer Science from Seoul National University in 2013. From 2002 to 2004, she joined Shinhan Financial Group as a developer in applied research. In 2010, she was with U.C. Berkeley as a researcher. From 2013 to 2016, she was with Soongsil University as a professor in the School of Electronic Engineering. Since 2016, she has been with Ajou University as a professor in DASAN University Colleage. Her research interests include social and computer networks, and security.

Cliff C. Zou received his BS and MS Degree from University of Science and Technology of China in 1996 and 1999, respectively. Then, he received the PhD Degree in Department of Electrical and Computer

Engineering from University of Massachusetts, Amherst, MA, in 2005. Currently, he is an associate professor at the Department of Computer Science, University of Central Florida. His research interests include computer and network security, network modelling and performance evaluation.

Soyoung Park received her BS, MS in Computer Science and PhD on Cryptology from Ewha Womans University, Korea in 1998, 2000 and 2006 respectively. Currently, she is an assistant professor at the Department of Internet and Multimedia Engineering, Konkuk University, Korea. Her research interests include cryptography, network security and privacy preserving ubiquitous computing.