Reputation-based Dissemination of Trustworthy Information in VANETs

V. Agate, A. De Paola, G. Lo Re, and A. Virga

In Proceedings of 20th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services – EAI MobiQuitous 2023
DOI: (to appear)

Publisher: Springer

Reputation-based Dissemination of Trustworthy Information in VANETs

Vincenzo Agate, Alessandra De Paola, Giuseppe Lo Re, and Antonio Virga

Department of Engineering, University of Palermo, Palermo, Italy,
{vincenzo.agate, alessandra.depaola, giuseppe.lore,
antonio.virga01}@unipa.it

Abstract. With the emergence of new vehicle communication paradigms such as Vehicle-to-Everything, the possibility of providing advanced services to drivers is becoming a reality. The immediate and targeted warning of dangers offers the opportunity to increase driving safety and make optimal use of the road infrastructure. However, communication reliability between vehicles, or worse, passenger safety, may be compromised by vehicles modified to spread false information or create disorder under coordinated malicious groups. Solutions currently adopted in similar scenarios include the use of Reputation Management Systems (RMS), which allow the reliability of received information to be estimated. However, classic centralized RMSs do not fit the distributed and dynamic nature of vehicular networks.

In this paper, a step is taken towards the design of a fully distributed event detection and dissemination system for VANETs, based on vehicle and data reputation, which does not rely on any fixed communication infrastructure. A new reputation model is proposed to reliably detect events and a new communication protocol is defined to disseminate information among vehicles, based on the population protocol model. The experimental evaluation performed on realistic vehicle routes demonstrates the feasibility of the proposed system and its ability to withstand orchestrated attacks, with a significant performance improvement over other state-of-the-art solutions.

Keywords: VANET, Reputation Management, Event Dissemination

1 Introduction

Intelligent Transport Systems (ITS) have the potential to significantly improve road safety. The emerging Vehicle-to-Everything (V2X) communication paradigm promises to achieve such a goal by enabling easy exchange of information between vehicles, between vehicles and infrastructure, and between vehicles and vulnerable road users (VRUs) such as pedestrians and cyclists [1]. The pervasive use of this technology would pave the way for the implementation of smart services, such as congestion analysis, traffic management, collision avoidance, cooperative driving and comfort/infotainment applications [2], which could have a significant impact even on road safety, reducing both the severity and the number of accidents.

Two communication paradigms exist to regulate the flow of information between vehicles in a vehicle network: Vehicle-to-Infrastructure (V2I) and Vehicle-to-Vehicle (V2V). The first model uses roadside units (RSUs), and there may be difficulties in its realisation for environmental or economic reasons. On the contrary, V2V networks only exploit direct communication between vehicles, thus being independent of any pre-existing physical infrastructure. However, the quality of services provided through this paradigm can be negatively affected by several physical factors, such as vehicle speed, high traffic density and the presence of physical obstacles that lead to communication reliability problems. Furthermore, since centralized quality control is not possible, false information could be spread due to faulty sensors or, even worse, information could be conveniently falsified by malicious parties to cause damage to the system.

To address this problem, trust and reputation-based systems may be the suitable solution to assess the reliability of data exchanged through collaborative vehicle activity [3, 4]. Recent solutions proposed in the literature make use of trusted entities, complex authentication schemes and certificates, all of which are incompatible with the dynamic and distributed nature of the scenario under consideration.

This paper proposes a new solution to perform reliable event distribution, suitable for both V2V and V2X contexts. The proposed solution adopts a new communication protocol, based on the population protocol model [5], a theoretical model designed to manage a set of autonomous agents that interact randomly to carry out distributed computations. The proposed solution is also based on an original reputation management system to improve the overall quality of information (QoI) through timely analysis of vehicle reputation and filtering of distributed false messages. To this end, the system uses two different metrics to estimate both the reliability of an event on the basis of the knowledge derived from the information reported by all the interacting vehicles, and to estimate the reputation of the individual vehicles, in order to appropriately weight the information received. The experimental evaluation shows that the proposed solution is resistant to various types of attacks organised by malicious user groups and achieves better performance than recent solutions proposed in the literature.

The main contributions of the proposed work can be summarized as follows:

- A new fully distributed event detection and dissemination system for VANET that does not rely on any fixed communication infrastructure;
- A three-tier architecture that leverages a reputation management module to evaluate the reliability of vehicles and information received, and leverages the population protocol paradigm for event dissemination;
- An extensive experimental validation performed with a dataset containing realistic vehicle tracks over a real area, allowing the performance of the proposed system to be compared with a state-of-the-art system.

The rest of the paper is organized as follows: Section 2 presents related works and describes their main limitations; the proposed reputation model and the event diffusion protocol are described in Section 3; the experimental evaluation is presented in Section 4. Final considerations and conclusions are given in Section 5.

2 Related Work

Some of the main goals of ITS include improving road safety through Vehicular ad hoc networks (VANETs), reducing traffic accidents and residual hazards by enabling cooperation between vehicles for timely information exchange. However, this scenario opens the door to the spread of false information from noisy sensors, or worse, from malicious entities intent on causing inconvenience or damage to the system. In fully distributed environments, an increasingly common solution is to use trust and reputation techniques to detect untrustworthy data or malicious users [6,7]. The basic idea behind this type of solution is to use the history of past interactions to assess the trustworthiness of information shared by participants. There are numerous application scenarios in which reputation-based approaches have been used, such as IoT, e-commerce, participatory sensing [8], etc., but implementation in the context of vehicular networks is even more challenging due to the dynamic nature of the network and the volatility of the information processed.

The authors in [9] proposed a trust-based model to protect V2X communications against internal attacks, by using vehicle-referenced adaptive weights to filter recommendations. However, this solution relies on the presence of RSUs, which are not always available, and suffers from a progressive degradation of the QoI as one moves away from it.

Some works in the literature address the problem of estimating vehicle reputation even in the absence of RSUs, i.e. in a V2V scenario, but, given the highly dynamic nature of VANETs, which are characterized by unpredictable topologies due to the highly variable speeds of their nodes , this goal poses many challenges. Some solutions, such as [10], propose models that use only physical characteristics to evaluate nodes and the information they share, such as packet delivery rate (PDR) and average delivery delay (ADD). However, such systems fail to detect attackers who spread poor quality or false information.

The Attack-resistant Trust Management (ART) scheme [11], one of the first in the VANET domain, has been proposed to assess the reliability of both nodes and messages. Data reliability is estimated based on data collected from multiple vehicles. To measure the reliability of a node, the system combines the estimated probability that the vehicle will perform its tasks and the probability that its recommendations are reliable.

To counter the potential impact of negative feedback from malicious vehicles, the authors of [12] propose REPLACE. This is a reliable recommendation scheme based on a platooning service, i.e., a driving model in which vehicles with common goals move cooperatively. In REPLACE, the intent is to recommend a reliable platoon leader to coordinate the platooning service. However, this model exploits a centralized reputation system to calculate scores using user feedback and cryptographic techniques such as public key cryptography and session key agreement between vehicles and RSUs, which require fixed static infrastructure. Similarly, the authors of [13] propose a scheme that exploits clustering techniques to elect a cluster head (CH) responsible for sending trusted information into the network. However, the main drawback of this approach is the possibility of electing dishonest CHs when the majority of nodes are dishonest [14]. The study presented in [15] suggests a scheme to calculate trust between entities, identify malicious nodes, and disseminate this information in a network. Malicious vehicles are excluded from the network using different trust metrics, such as event-based trust, direct trust, and indirect trust. The dissemination of overall trust values is done through the periodic exchange of beacon messages. However, the attack model assumed by the authors, i.e., that malicious vehicles always exhibit malicious behavior, is unlikely to be true in a real-world context [16].

The authors of [17] propose a reputation management system called MARINE, with the purpose of detecting malicious nodes that launch man-in-the-middle attacks. The MARINE trust model works in two steps to assess inter-vehicle trust. First, it evaluates the sender node to determine its trustworthiness. This is done through previous interactions and recommendations from neighboring vehicles. Second, once node-centric trust is calculated, the received data is evaluated in three different dimensions: information quality, node's ability to forward messages, and neighbors' opinions. Data from the sender node is accepted only if node-centric and data trust are successfully calculated. Otherwise, the evaluating node will discard the data. MARINE relies on both vehicles and RSUs to compute the overall trust in the sender and the received information.

One of the most recent works on reputation management for vehicular networks has been proposed by the authors of [18]. In this work, the authors propose a distributed RMS that evaluates the behavior of nodes participating in the network as the result of direct interaction and through the recommendation values obtained from neighborhood. In order to correlate the two trust values (direct and indirect), the system uses a coefficient that takes into account qualitative attributes such as familiarity, similarity and timeliness. This allows different weights to be assigned to both trust values. Finally, to detect and identify a malicious node, a trust threshold mechanism is introduced, which, based on specific rules, determines whether or not to exclude the node under investigation from the network.

An in-depth discussion of recent work in which the assessment of the quality of shared information is driven by trust in VANETs can be found in [19].

Some of the approaches recently proposed [14,17] in the literature base information quality control on fixed infrastructural elements or trusted entities that may not always be available, on the contrary, in the solution proposed here the diffusion of events is obtained through a population protocol in a completely distributed manner. Furthermore, information is disseminated on messages to other vehicles only after passing through a check by a reputation management module. By exploiting the proposed layered architecture, the reputation module

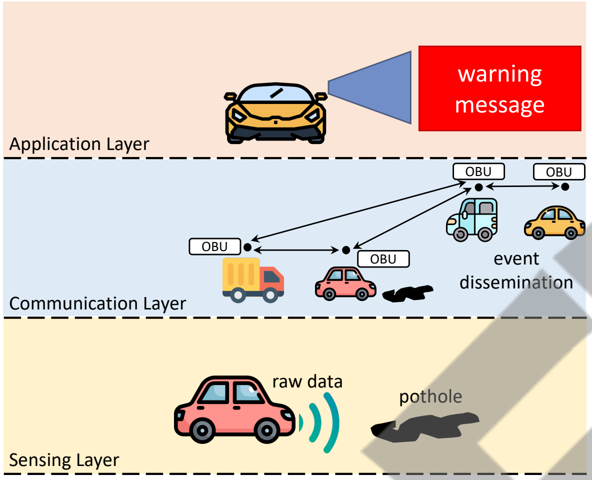


Fig. 1: The multi-layer architecture of the system.

is able to exploit not only the direct experiences of event perception [20], but also the appropriately weighted reported information.

3 The Multi-Layer Vehicular Architecture

The system proposed here exploits a three-layer architecture structured as shown in Figure 1.

At the lowest layer, the *Sensing Layer* (SL) performs the task of detecting events during vehicle motion. To fulfill this role, the *Sensing Layer* exploits on-board sensors (e.g., cameras, proximity sensors, accelerometers, gyroscopes, air quality sensors, GPS, etc.) that interact with the surrounding environment. At this level, the raw data collected by the sensors as the vehicle moves is passed to a data fusion module, which is responsible for analyzing and classifying events of interest, such as the presence of traffic congestion, potholes [21], accidents and so on. The output of the *Sensing Layer* is a list of events that are sent to the higher layer, the *Communication Layer* (CL), which has two main functions: to spread the knowledge of events directly sensed by the vehicle and to receive events reported by other vehicles with which it has managed to communicate. The functions implemented by the *Communication Layer* allow vehicles to know events promptly thanks to information obtained by other vehicles. To achieve a reliable event dissemination algorithm, the quality of the spread information must not be underestimated. Since no centralized authority can be utilized, the dissemination of false information opens the door to possible attack scenarios.

To maintain control over the quality of disseminated events, a distributed reputation and trust module is employed. This ensures that only reliable events are shared with other vehicles. To this end the *Communication Layer* uses a

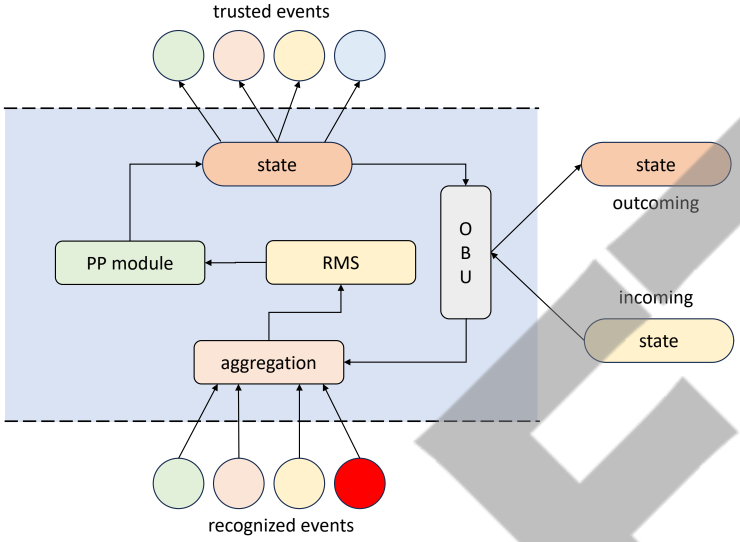


Fig. 2: Internal structure of the Communication Layer.

communication protocol based on Population Protocols, a communication model originally designed to enable sensor networks with limited resources to achieve a common view of their environment. The vehicle cooperation mechanism, carried out by the joint action of the reputation module and the population protocol, is essential for vehicles to obtain information about events not in their proximity and in advance, enabling them to plan appropriate actions to achieve their goals.

Although not all vehicles currently have the ability to benefit from hardware such as OBUs and sensors, which allow them to freely share data to participate in ITS services, it is possible that in the near future manufacturers will equip all vehicles on the road with such tools. Additionally, numerous efforts are underway to make these technologies available for older vehicles at an affordable cost [22].

On top of this architecture, the *Application Layer* (AP) receives the list of reliable events generated by the *Communication Layer* to provide specific advanced services aimed at improving the user's driving experience and safety (such as notification of nearby incidents, the ability to plan actions based on events, etc.).

The core of this system is the Communication Layer, whose main components, shown schematically in Figure 2, are described in the remainder of this section. This layer has an internal state that summarizes events detected or inferred from messages sent by other vehicles. This information is shared with other vehicles through the on-board units (OBUs), which are also responsible for collecting incoming messages. Following the rules of the proposed population protocol, the received messages are subjected to a filtering process based on a reputation model, and the resulting information contributes to the state update.

3.1 State Model

In the system proposed here, communication between vehicles is performed according to the rules of a Population Protocol (PP). Each vehicle follows the protocol and interacts with other vehicles in its communication range. During an interaction between source and destination vehicles, a message containing a triplet of values is sent:

$$m = \{v_{id}, s, ts\}\,, \tag{1}$$

where v_{id} is a unique identifier for each vehicle, and ts is the timestamp at which the message was sent. The most important part of the message is the state s, which contains two lists:

$$s = \{E_T, E_U\}, \tag{2}$$

where E_T is a list of events considered reliable and E_U is a list of events considered unreliable. The content of each list is determined taking into account both direct interaction with the environment and information reported by other vehicles. The details of how the two lists are populated are discussed later in this section.

Each event ϵ_i contained in one of the two lists of the state s is a tuple of values composed as follows:

$$\epsilon_i = \begin{cases} class_i \\ (x_i,\ y_i) \\ ts_i \\ v_{id} \end{cases} \tag{3}$$

where $class_i$ is an attribute indicating the type of event (e.g., accident, pothole, etc.), the pair $(x_i,\ y_i)$ indicates an approximation of the geographic coordinates relative to the event, ts_i indicates the time at which the event was detected, and finally v_{id} indicates the identifier of the vehicle that detected the event.

3.2 Aggregation of Events

Before assessing the trust level of events and the reputation of vehicles, events are grouped according to their class and geographical coordinates. Since different vehicles may perceive the same event in slightly different geographical locations due to poor accuracy or noise, it is crucial that these events are considered to be the same actual event. Figure 3 helps to understand the problem just discussed.

The proposed solution involves a clustering step implemented through the DBSCAN algorithm [23], which is designed to detect clusters, even of different densities, in large spatial datasets. The choice of this clustering algorithm over others is motivated by the fact that it does not require knowing the number of clusters in advance and can work with clusters of any distribution and shape. Its operation is governed by two parameters, namely the maximum allowed distance eps between two points in the same cluster and the minimum allowed size $MinPts$ of a cluster. In the considered scenario they represent the maximum

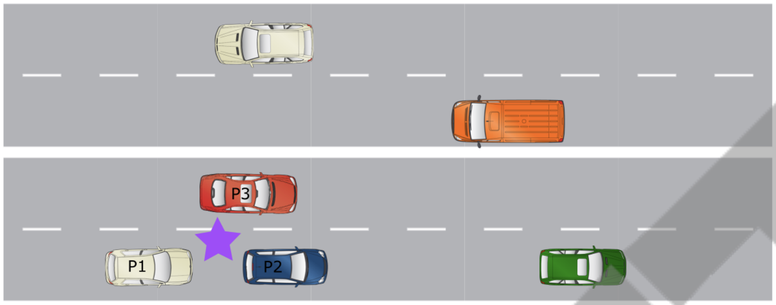


Fig. 3: Example scenario where a set of events reported by different vehicles refer to a single real event. The clustering phase is necessary to aggregate slightly different views within the system.

distance in terms of geographic coordinates between two events that should be considered as the same real event and the minimum number of events to be able to form a cluster respectively. Since it is not possible to determine a priori how many times the same event will be perceived, and even a single event may be sufficient to identify a cluster by itself, the value $MinPts$ is assumed to be $MinPts = 1$, thus avoiding single actual events as noise points. The *eps* parameter is set by considering as a reference the maximum distance at which the vehicle sensors are able to obtain raw measurements, plus a tolerance margin of 10% as a fair trade-off to avoid the presence of cluster merging or fragmentation.

3.3 Reputation Model

To quickly propagate event information, each vehicle sends its neighborhood a list of events it considers trustworthy and untrustworthy. Since reported event lists may contain unreliable information, the Reputation Management Subsystem (RMS) is responsible for estimating the level of trust in reported events whenever one is received from a vehicle. To do this, the RMS performs three basic steps: computing the local trust of events (I), estimating the reputation of vehicles (II), and finally computing the trust of received events (III).

Local Trust of Events. Once the RMS receives the list of all aggregated events, it proceeds to calculate the local trust. The different events are now distinguishable by a tag specifying their identifier, called $cluster_{id}$, and through this the RMS can actually count the number of times it has been reported by a vehicle as a trusted and untrusted event. The local trust value $lt(\epsilon_i)$ for each event ϵ_i is calculated through the following equation:

$$lt(\epsilon_i) = \frac{m(\epsilon_i) - k(\epsilon_i)}{n}, \tag{4}$$

with $m(\epsilon_i), k(\epsilon_i) \in [0, n]$ and $0 < m(\epsilon_i) + k(\epsilon_i) \leq n$. The value $m(\epsilon_i)$ represents the number of vehicles that consider the i-th event as trustworthy (i.e., ϵ_i is contained in the E_T list), while $k(\epsilon_i)$ is the number of vehicles that consider the i-th event as untrustworthy, (i.e., ϵ_i is contained in the E_U list). Finally, n is the number of vehicles that sent their state during the last time interval. Given Equation 4, the values of $lt(\epsilon_i)$ are in the range $[-1, 1]$.

Vehicle Reputation. Using the trust values of the individual events $lt(\epsilon_i)$ obtained in the previous step, it is now possible to assess the trustworthiness of neighbouring nodes. The idea behind the new score is to evaluate the amount of reliable information received by each of the participants with whom the vehicle interacted.

Assuming that each vehicle has a limited amount of memory available, only the events contained in a time window of fixed size, W, are considered. In order to make the reputation values of the k-th vehicle resistant to sudden fluctuations, the history of received messages is taken into account. For this reason, each received message is associated with a trust value calculated as follows:

$$t_m(m_j^k) = \frac{\sum\limits_{\epsilon_i \in m_j^k} lt(\epsilon_i)}{\left| m_j^k \right|}, \tag{5}$$

where m_j^k represents the j-th message received by the vehicle k.

After obtaining the trust values of each message, the RMS proceeds to calculate the reputation of the vehicle k, called $r(v_k)$, by averaging the trust values of the messages received by it:

$$r_k^t = \alpha * t_m(m_j^k) + (1 - \alpha) * r_k^{t-1}. \tag{6}$$

Note that the above equation uses the exponentially weighted moving average (EWMA), which, as the smoothing factor α varies, allows us to weigh the most recent observation more or less heavily against past history.

Trust of Received Events. After the calculation of the reputation values, the RMS estimates the trust values for the events contained in the messages received in the current step. This phase is essential for the events received by the OBU to reach the PP, which, based on the trust values, will decide to place the events in one of the two lists contained in the state to be shared.

The trust $t(\epsilon_i)$ of the i-th event is calculated through the following equation:

$$t(\epsilon_i) = \frac{\sum\limits_{k \in K} r_k^t - \sum\limits_{z \in Z} r_z^t}{n}, \tag{7}$$

where K is the set of nodes that reported that the event ϵ_i was reliable, while vehicles in Z reported the opposite and $n = |K| + |Z|$.

Algorithm 1: Transition Function $\delta(\cdot)$ upon receiving an internal state

```
Input  : states of the receiving node ($s_i$), states of the sending node ($s_j$)
Output: Updated $s_i$ to $s'_i$
for $event \in s_j.E_T$ do
    if $event \not\subset s_i.E_T$ and $event \not\subset s_i.E_U$ then
        $s_i.E_T.append(event)$;
    else if $event \subset s_i.E_U$ then
        $s_i.E_U.remove(event)$;
        $s_i.E_T.append(event)$;
for $event \in s_j.E_U$ do
    if $event \in s_i.E_T$ then
        $s_i.E_T.remove(event)$;
        $s_i.E_U.append(event)$;
```

3.4 Diffusion model

The goal of the PP module is to ensure that the status of the vehicle is updated with information acquired both during communication with neighboring nodes and with events obtained from the lower Sensing Layer [24, 25]. The proposed diffusion model exploits a unidirectional PP [5], which unlike the basic PP, where the update of states occurs only after two agents are mutually synchronized (i.e., after both have exchanged states), the update occurs upon receipt of a message. This feature is specifically designed for VANETs, since node mobility and other phenomena typical of wireless communications, such as fading, could compromise the symmetry of communication.

The Population Protocol proposed here receives the following input paramenters:

- An alphabet of possible initial values Σ, that contains only the null symbol $\varnothing$. In this way, the input function ι initializes the state of each node to a single base configuration common to all nodes. More formally:

$$s : \iota(\varnothing) = \{E_T, E_U\} \ where \ E_T, E_U = \emptyset \tag{8}$$

- An output function ω that, given an input state s, generates a set of trustworthy and untrustworthy ϵ_i events contained in two different lists.
- A transition function δ, used to update the state of the receiving node, defined as follows:

$$\delta(s_i, s_j) = (s'_i, s'_j). \tag{9}$$

  In the above equation s_i and s_j represent the states of the receiving node and sending node before the update, while s'_i and s'_j are the states after the update. Considering that the chosen PP is unidirectional, the state of the sender node remains unchanged and only the state of the receiver node undergoes a change, in other words $s_j = s'_j$.

Algorithm 2: Transition Function $\delta(\cdot)$ upon receiving an external state

```
   Input  : states of the receiving node ($s_i$), states of the sending node ($s_j$), id
            of the vehicle that is performing the calculation ($myId$), threshold
            high ($\theta_h$), threshold low ($\theta_l$)
   Output: Updated $s_i$ to $s'_i$
1  for $event \in s_j.E_T \cup s_j.E_U$ do
2      if $Trust(event) > \theta_h$ then
3          if $event \not\subset s_i.E_T$ and $event \not\subset s_i.E_U$ then
4              $s_i.E_T.append(event)$;
5          if $event \in s_i.E_T$ then
6              $Update(event)$;                    ▷ Update timestamp and vehicle id
7          if $event \in s_i.E_U$ and $s_i.E_U.event.v_{id} \neq myId$ then
8              $s_i.E_U.remove(event)$;
9              $s_i.E_T.append(event)$;
10     if $Trust(event) < \theta_l$ then
11         if $event \not\subset s_i.E_T$ and $event \not\subset s_i.E_U$ then
12             $s_i.E_U.append(event)$;
13         if $event \in s_i.E_U$ then
14             $Update(event)$;                    ▷ Update timestamp and vehicle id
15         if $event \in s_i.E_T$ and $s_i.E_T.event.v_{id} \neq myId$ then
16             $s_i.E_T.remove(event)$;
17             $s_i.E_U.append(event)$;
```

The transition function $\delta(\cdot)$ has two different behaviors depending on the source of the received state s_j. In particular, the state s_j could come from a node with which the vehicle has interacted or from the *Sensing Layer*. In both cases, the internal state must be updated. Algorithms 1 and 2 summarize the transition function for both scenarios.

According to Algorithm 1, i.e. the case where s_j comes from the underlying layer of the architecture, the events ϵ_i perceived from the vehicle's environment are considered fully trusted. The action performed by the transition function in such a case is to adjust the internal state s_i by placing the events in the lists E_T and E_U, checking for consistency with the knowledge base.

When s_j is the result of exchanging messages with another vehicle, the transition function follows the behavior specified in Algorithm 2. In this scenario, the trust values associated by the node to the received events assumes a key role in correctly updating the s_i state. After setting two thresholds θ_h and θ_l, the first to discriminate trusted events and the second for untrusted events, the algorithm proceeds by analyzing the received events one by one, comparing the trust with the thresholds. As can be seen from the algorithm, the part of pseudocode from line 2 to line 9 concerns events whose confidence is greater than θ_h. In this case, the transition function evaluates whether the event should be

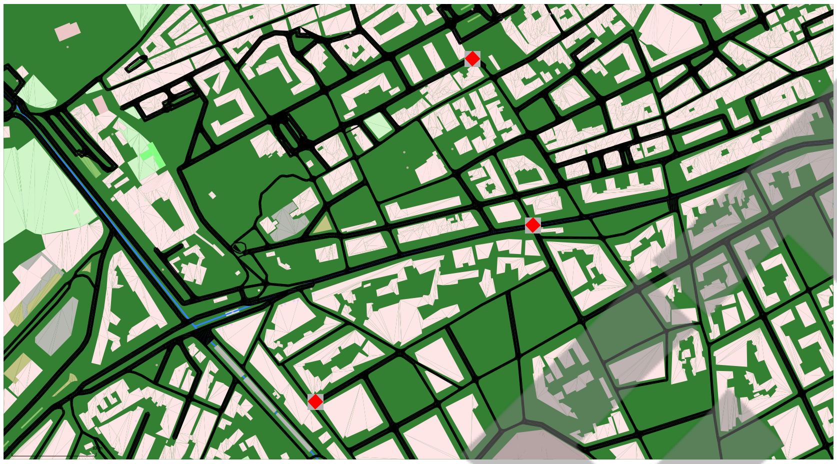


Fig. 4: Map of the test area.

placed in E_T if it was not previously known, or if it is in the list of untrusted events E_U, it should be moved if and only if the information does not conflict with that obtained by its own sensory apparatus. The logic used is that direct experiences always overrides referred information.

Finally from line 10 to line 17, the transition function performs the opposite behavior for events whose trust is below the θ_l threshold.

4 Experimental Evaluation

This section shows the results of the experimental evaluation of the proposed solution, demonstrating that it is well suited for the dissemination of truthfull events in VANETs, and is also resilient to the threat of security attacks by vehicles opportunistically orchestrated to undermine the system.

In addition, the performance of the proposed system in terms of accuracy, precision, recall, and F1-score are compared with the performance of a state-of-the-art technique.

4.1 Simulation Environment

The experimental evaluation has been performed by using the open source framework VEINS [26], which is based on two simulators, SUMO [27], a road traffic simulation suite, and OMNET++ [28], a C++-based simulation library that is suitable for creating network simulators. The proposed model can be employed in a variety of road scenarios, regardless of the network topology. The experiments reported in the following were conducted in urban environments simulated from real maps.

4.2 Experimental setting

The OpenStreetMap tool was used to create the simulation dataset, and the resulting map covers an area of the university campus of the city of Palermo. Part of the map is visible in Figure 4, and the size of the working area is 6 Km^2.

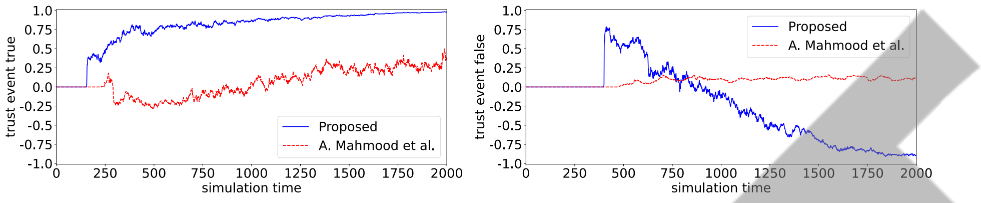
Communication was made more realistic by adding obstacles such as buildings using SUMO's polyconvert script. The communication radius is set to 80 m for each of the vehicles and a message exchange frequency is 5 Hz. The maximum useful distance for recognition of directly perceived events was also set at 3 m for each node, and the speeds of the vehicles in the map depend on the actual speed limits imposed by the roads, which range from 7 m/s to 13 m/s. In order to maintain a proper balance between new reputation values and past history, the α value is set to 0.5 [29].

The red-colored indicators in Figure 4 represent the events of interest for the scenario considered here (e.g., construction sites, hazards, accidents, etc.). The locations of the events were chosen from all possible intersections of the roads, extracting them according to a Gaussian distribution with mean 0 and variance 1. Each simulation was run for the duration of about half an hour (2000 s).

To demonstrate the effectiveness of the proposed system, its performance was compared with one of the best performing systems in recent literature, which does not use a fixed infrastructure for reputation calculation and information dissemination [18]. The authors propose a system to assess the reputation of vehicles participating in the network by exploiting both direct and indirect trust in order to exclude a possible untrusted node. To do this, they implement a thresholding mechanism that gives a vehicle multiple opportunities to improve its behavior before it is removed from the sharing network. To make a fair comparison between the two systems, different experimental runs were performed under the same conditions, replacing the proposed RMS subsystem with their proposed reputation mechanism and leaving the information dissemination system based on the Population Protocol unchanged.

Model of malicious vehicle. In order to test the system under complex though realistic attack scenarios, a set of vehicles with malicious behavior are generated in the map. To ensure that the attack scenario is even more effective, such vehicles are distributed in the map to form evenly distributed outbreaks.

The behaviour of an attacking vehicle consists of adding false information in the messages it disseminates to other vehicles, such as events that do not really exist on the map. In order to make the attack even more insidious, the attacker injects small amounts of false information into messages containing the majority of trustworthy information [30, 31]. As a result, the attacker is unlikely to be completely detected while maintaining an average reputation value. In the experiments described below, there will be between 10% and 30% malicious vehicles out of a total of 100 vehicles, and each malicious vehicle spreads false information about one event out of 8 authentic events on the map. The attacks start at simulation time 400.



(a) (b)

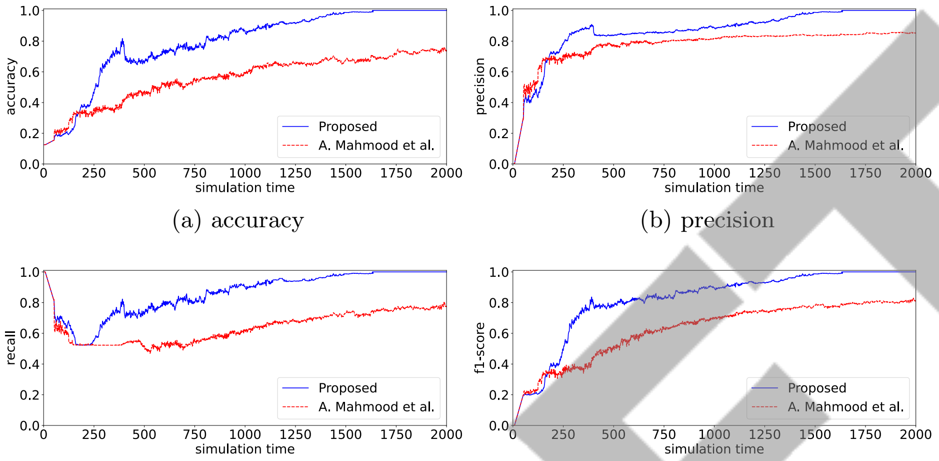
Fig. 5: Comparison of the trend of trust values of a good event (a) and a malicious event (b) between the proposed system and the solution adopted as benchmark.

Since the range of possible trust values is $[-1, 1]$, such interval is divided in the following three regions: trusted, untrusted, and uncertain region. The thresholds values to discriminate an event as trusted or untrusted were set at $\theta_h = 0,33$ and $\theta_l = -0,33$, so that the range of possible confidence values is divided equally among the three labels (i.e., trustworthy, uncertain, and untrustworthy).

4.3 Experimental Results

The first experiment, shown in Figure 5, is useful to compare the basic operation of the system proposed here against the solution proposed in [18]. Specifically, Figure 5-a shows the trend of the average trust value of an true event, while Figure 5-b shows the trend of a false event propagated by the attacking nodes. In this first experiment there are 100 vehicles in the map of which 5% spread false information. As the left curve shows, both systems correctly estimate the real event, with a trend toward the maximum trust value as simulation time increases, although the proposed approach reaches the maximum trust value faster. Figure 5-b shows the trend of the trust value of a malicious event spread simultaneously by the attacking vehicles at time 400. By the curve in blue, it is possible to observe that the average trust estimated by vehicles exploiting the proposed system is initially affected by the false information disseminated by attackers initially considered trustworthy. After the initial phase, the system gradually recognizes false information, as confirmed by the decreasing trend of trust. Surprisingly, the system proposed in [18] initially appears to be unaffected by the spread of false information. However, the slightly increasing trust trend suggests that the system fails to isolate the false information, and suffers the nodes attack. Effectively, the attack is particularly insidious for both systems given that according to the previously described attack model the misinformation, i.e., the false event, is anyway spread along with a number of correct information that allows the attacking nodes to survive in both systems. This demonstrates the remarkable false-event discrimination capability of the approach proposed here.

The second set of experiments aims to quantitatively compare the performance of both systems in correctly detecting events. Since malicious event detection can be considered as a binary classification problem, where an event can



(a) accuracy

(b) precision

(c) recall

(d) f1-score

Fig. 6: Comparisons of the trends of evaluation metrics of the proposed model against the RMS proposed in [18].

be either true or false, according to [32], the performance of both systems can be evaluated using the following well known metrics: *accuracy*, that is the ratio of correct predictions to all predictions made, *precision*, which is the ratio of correct positive predictions to all positive predictions made, *recall*, which is the ratio of correct positive predictions to all predictions that must be positive, and *F1_score*, which is the harmonic mean between *precision* and *recall*. To solve the problem of the large imbalance between the class of actual events and the class of malicious events, a weighted definition of these metrics was adopted.

Figure 6 shows the trends of the four metrics examined. From Figure 6-a showing the accuracy trend, it is possible to see the strong initial growth due to the bootstrap phase. During this phase all nodes entered into the network begin to build their knowledge base from scratch. Note how, from simulation time 250, the proposed system outperforms the classification capabilities of the competitor. Starting at time step 400, the clique attack of 10% of the total nodes begins. The attack is appropriately orchestrated so that all attackers simultaneously spread the same false event.

The performance of the system actually degrades slightly, although it remains above its competitor. Although the attack phase never ends until the end of the experiment, the proposed system succeeds in reaching the maximum accuracy value confirming its ability to detect and isolate false information spread by attackers. The Precision, Recall and F1-score curves also confirm the superiority of the proposed system.

The last experiment, shown in Figure 7, aims to evaluate the cumulative performance of the 4 metrics on the two systems as the percentage of attacking

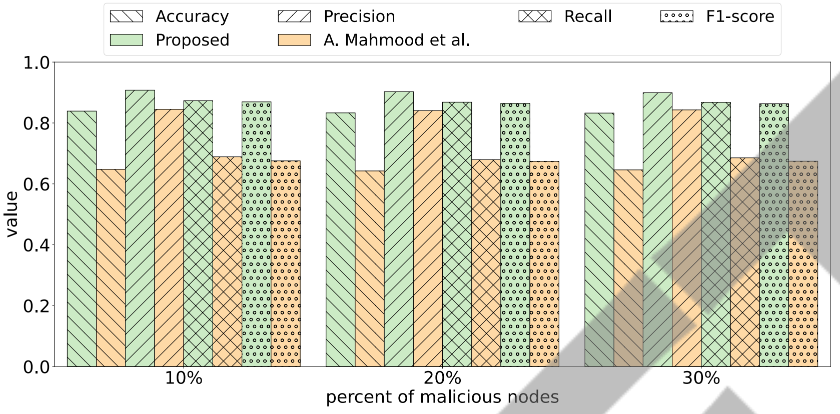


Fig. 7: Comparison of systems performance as the percentage of attackers on the network changes.

nodes varies from 10% to 30% of the total vehicles. All metrics show imperceptible performance degradation as the number of attackers increases, demonstrating the effectiveness of both solutions. In all attack configurations however, the system proposed here consistently outperforms the second system by achieving the largest gap in F1-score value (about 30% better than its competitor) and the smallest gap between precision values (about 10% better than its competitor).

5 Conclusion

This paper presents a novel solution for the dissemination of reliable information within VANETs, which can recognise malicious vehicles and exclude false events spread to cause damage to the network.

In the proposed system, trust in the event is estimated by exploiting two separate metrics, namely the local trust of the event and the reputation of the vehicles sharing information on the same event. The former provides a preliminary estimate of the trustworthiness of an event based on knowledge derived from information reported by other vehicles, while the latter depends on the reliability of vehicles in the network, i.e. their reputation.

Unlike other approaches proposed in recent literature, which rely on an infrastructure or trusted entities that may not always be available, the proposed approach employs a fully distributed communication method based on the population protocol model. In the proposed solution, event information is shared only after being filtered based on the reputation of communicating vehicles, thus ensuring the reliability of the event detection algorithm.

The solution proposed here was compared with a state-of-the-art system and the results of the experiments, performed on real traces, demonstrated its superiority with respect to all metrics considered, resulting in an improvement of up to 30% in the F1-score.

Acknowledgment

This research is partially funded by the S6 Project (PO FESR Sicilia 2014/2020) and partially funded by the European Union - PON Ricerca e Innovazione 2014-2020 - DM 1062/2021.

References

1. Yoshizawa, T., Singelée, D., Muehlberg, J.T., Delbruel, S., Taherkordi, A., Hughes, D., Preneel, B.: A survey of security and privacy issues in v2x communication systems. ACM Comput. Surv. 55(9) (jan 2023), https://doi.org/10.1145/3558052
2. Singh, S., Agrawal, S.: Vanet routing protocols: Issues and challenges. 2014 Recent Advances in Engineering and Computational Sciences (RAECS) pp. 1–5 (2014)
3. Agate, V., De Paola, A., Lo Re, G., Morana, M.: A platform for the evaluation of distributed reputation algorithms. In: 2018 IEEE/ACM 22nd International Symposium on Distributed Simulation and Real Time Applications (DS-RT). pp. 1–8. IEEE (2018)
4. Huang, X., Yu, R., Kang, J., Zhang, Y.: Distributed reputation management for secure and efficient vehicular edge computing and networks. IEEE Access 5, 25408–25420 (2017)
5. Aspnes, J., Ruppert, E.: An introduction to population protocols. Middleware for Network Eccentric and Mobile Applications pp. 97–120 (2009)
6. Agate, V., De Paola, A., Lo Re, G., Morana, M.: Vulnerability Evaluation of Distributed Reputation Management Systems. In: InfQ 2016 - New Frontiers in Quantitative Methods in Informatics. pp. 1–8. ICST, ICST, Brussels, Belgium (2016)
7. Agate, V., De Paola, A., Lo Re, G., Morana, M.: Dress: A distributed rms evaluation simulation software. International Journal of Intelligent Information Technologies (IJIIT) 16(3), 1–18 (2020)
8. Timilsina, A., Khamesi, A.R., Agate, V., Silvestri, S.: A reinforcement learning approach for user preference-aware energy sharing systems. IEEE Transactions on Green Communications and Networking (2021)
9. Alnasser, A., Sun, H., Jiang, J.: Recommendation-based trust model for vehicle-to-everything (v2x). IEEE Internet of Things Journal 7(1), 440–450 (2019)
10. Tan, S., Li, X., Dong, Q.: A trust management system for securing data plane of ad-hoc networks. IEEE Transactions on Vehicular Technology 65(9), 7579–7592 (2015)
11. Li, W., Song, H.: Art: An attack-resistant trust management scheme for securing vehicular ad hoc networks. IEEE transactions on intelligent transportation systems 17(4), 960–969 (2015)
12. Hu, H., Lu, R., Zhang, Z., Shao, J.: Replace: A reliable trust-based platoon service recommendation scheme in vanet. IEEE Transactions on Vehicular Technology 66(2), 1786–1797 (2016)
13. Mahmood, A., Butler, B., Zhang, W.E., Sheng, Q.Z., Siddiqui, S.A.: A hybrid trust management heuristic for vanets. In: 2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops). pp. 748–752 (2019)
14. Ahmad, F., Kurugollu, F., Kerrache, C.A., Sezer, S., Liu, L.: Notrino: A novel hybrid trust management scheme for internet-of-vehicles. IEEE Transactions on Vehicular Technology 70(9), 9244–9257 (2021)

15. Kerrache, C.A., Lagraa, N., Calafate, C.T., Cano, J.C., Manzoni, P.: T-vnets: A novel trust architecture for vehicular networks using the standardized messaging services of etsi its. Computer Communications 93, 68–83 (2016)
16. Agate, V., De Paola, A., Gaglio, S., Lo Re, G., Morana, M.: A framework for parallel assessment of reputation management systems. In: Proceedings of the 17th International Conference on Computer Systems and Technologies 2016. pp. 121–128 (2016)
17. Ahmad, F., Kurugollu, F., Adnane, A., Hussain, R., Hussain, F.: Marine: Man-in-the-middle attack resistant trust model in connected vehicles. IEEE Internet of Things Journal 7(4), 3310–3322 (2020)
18. Mahmood, A., Sheng, Q.Z., Zhang, W.E., Wang, Y., Sagar, S.: Toward a distributed trust management system for misbehavior detection in the internet of vehicles. ACM Trans. Cyber-Phys. Syst. 7(3) (jul 2023), https://doi.org/10.1145/3594637
19. Hussain, R., Lee, J., Zeadally, S.: Trust in vanet: A survey of current solutions and future research opportunities. IEEE transactions on intelligent transportation systems 22(5), 2553–2571 (2020)
20. Agate, V., Ferraro, P., Gaglio, S.: A cognitive architecture for ambient intelligence systems. In: AIC. pp. 52–58 (2018)
21. Agate, V., Concone, F., Ferraro, P.: A resilient smart architecture for road surface condition monitoring. In: The Proceedings of the International Conference on Smart City Applications. pp. 199–209. Springer (2021)
22. Mistareehi, H.: Message dissemination scheme for rural areas using vanet (hardware implementation). In: 2021 Twelfth International Conference on Ubiquitous and Future Networks (ICUFN). pp. 120–125 (2021)
23. Ester, M., Kriegel, H.P., Sander, J., Xu, X., et al.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: kdd. vol. 96, pp. 226–231 (1996)
24. Bordonaro, A., Concone, F., De Paola, A., Lo Re, G., Das, S.K.: Modeling efficient and effective communications in vanet through population protocols. In: 2021 IEEE International Conference on Smart Computing (SMARTCOMP). pp. 305–310 (2021)
25. Bordonaro, A., De Paola, A., Lo Re, G.: Vpp: A communication schema for population protocols in vanet. In: 2021 20th International Conference on Ubiquitous Computing and Communications (IUCC/CIT/DSCI/SmartCNS). pp. 11–18 (2021)
26. Sommer, C., German, R., Dressler, F.: Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis. IEEE Transactions on Mobile Computing (TMC) 10(1), 3–15 (January 2011)
27. Lopez, P.A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y.P., Hilbrich, R., Lücken, L., Rummel, J., Wagner, P., Wießner, E.: Microscopic traffic simulation using sumo. In: 2018 21st international conference on intelligent transportation systems (ITSC). pp. 2575–2582. IEEE (2018)
28. Varga, A., Hornig, R.: An overview of the omnet++ simulation environment. In: 1st International ICST Conference on Simulation Tools and Techniques for Communications, Networks and Systems (2010)
29. Agate, V., De Paola, A., Lo Re, G., Morana, M.: A simulation software for the evaluation of vulnerabilities in reputation management systems. ACM Transactions on Computer Systems (TOCS) 37(1-4), 1–30 (2021)
30. Crapanzano, C., Milazzo, F., De Paola, A., Lo Re, G.: Reputation management for distributed service-oriented architectures. In: 2010 Fourth IEEE International

Conference on Self-Adaptive and Self-Organizing Systems Workshop. pp. 160–165. IEEE (2010)

31. Agate, V., De Paola, A., Lo Re, G., Morana, M.: A simulation framework for evaluating distributed reputation management systems. In: Distributed Computing and Artificial Intelligence, 13th International Conference. pp. 247–254. Springer (2016)
32. Khatri, N., Lee, S., Mateen, A., Nam, S.Y.: Event message clustering algorithm for selection of majority message in vanets. IEEE Access 11, 14621–14635 (2023)