Simple Feedback Driven Accuracy Based Reputation Mechanism for IoV

Rohan Dahiya, Frank Jiang, and Robin Doss, *Senior Member, IEEE*

***Abstract*—Most applications of Internet of Vehicles (IoV) rely on collaboration between the nodes. There is therefore an implicit trust between each of these nodes, which is why false information being shared poses a challenge in IoV. To overcome this issue several mechanisms have been proposed for detection of false information and also for trust management in IoV, most of which employ reputation scores as an important factor. It is therefore important to have a robust and consistent mechanism that can calculate a reputation score for each node based on the accuracy of the information shared by it. Such a mechanism has been proposed in this paper. The proposed system first utilises the results of false message detection methods to generate and share feedback in the network, this feedback is then collected and filtered to remove potentially malicious feedback, which is then used to generate a dynamic reputation score for each node. The reputation system has been experimentally shown to have high accuracy in detection of nodes sending false information and is negligibly affected in the presence of spurious feedback.**

***Index Terms*—IoV, VANET, V2X, Reputation System, Trust Management.**

I. Introduction

Vehicular ad-hoc networks (VANETs) which have been around for decades are based on the principles of mobile ad-hoc networks (MANETs) being applied to the domain of vehicles and infrastructure. They consist of On Board Units (OBUs) in vehicles (peers/nodes) and Road Side Units (RSUs). Internet of Vehicles (IoV) is a concept that is derived from VANETs where each node and RSU is internet enabled and is capable of communicating with each other via dedicated short-range communication (DSRC). Numerous uses of IoV have been proposed from infotainment to traffic safety related applications such as traffic information sharing, emergency vehicle notification systems and collision avoidance systems.

5G, the fifth generation technology standard for cellular networks [1], began being deployed in 2019 and mid-band 5G (which offers much higher data transfer rates) were expected to be available in most metropolitan areas by 2020. In Europe, a consortium of companies, is helping to develop a 5G system architecture to provide optimized end-to-end vehicle to everything (V2X) connectivity called the 5GCAR project. V2X is an umbrella term that refers to a combination of more specific types of communication as V2I (vehicle-to-infrastructure), V2N (vehicle-to-network), V2V (vehicle-to-vehicle), V2P (vehicle-to-pedestrian), V2D (vehicle-to-device) and V2G (vehicle-to-grid). 5G can be expected to mitigate some of the limitations in IoV and pave the way for applications that would not have been plausible before.

Most application in IoV rely on cooperativeness of other nodes to function especially applications that are based on the exchange of information between nodes. False information being shared by malicious nodes can negatively affect the utility of these applications, as detailed in II-A. For this reason, false information detection systems have been proposed to enable vehicles to filter out the as much of it as possible and trust management systems are employed in Social Internet of Vehicles (SIoV) to quantify the reliability of information received. Some of these mechanisms have been touched upon in II-B and II-C.

Many of the aforementioned mechanisms employ reputation scores as an important factor, hence it is important to have a consistent mechanism that can calculate reputation scores for all nodes with high positive correlation to the overall accuracy of the respective nodes. Further, the system should be resistant to manipulation. The scope of this paper is to introduce a partly centralised reputation mechanism that aims to accurately estimate the accuracy of a node's messages based on feedback from other nodes while filtering out false feedback. The system is shown, experimentally, to have less than five percent error in estimation of the overall accuracy of a node, moreover, the system's robustness was evaluated in various scenarios of attempted system manipulation and was found to be foolproof with negligible degradation in performance up-to reasonable limits of manipulation. This system can be hybridised with earlier work as the reliable reputation score generated can be used as a factor in a trust management system or a false message detection system as many of these proposed systems, mentioned in subsequent sections, rely on reputation scores as a factor. It may also be incorporated in subsequent work that requires reputation scores for vehicles based on information shared by them as an input factor.

The rest of this paper is structured as follows: A review of related works on attacks in IoV, False information detection systems and trust management systems is presented in Sections II-A, II-B and II-C respectively. The proposed methods are described in Section III. The proposed methods were tested via software simulation, details on the experiments and the observations are presented in Section V and Section VI respectively, and the conclusion is contained in Section VII.

R. Dahiya was a visiting researcher at the School of Info Technology, Deakin University, Burwood, VIC 3125, Australia, and is a student at Vellore Institute of Technology, Vellore, TN 632014, India. e-mail: rohan-dahiya@outlook.in

F. Jiang and R. Doss are with School of Info Technology, Deakin University, Geelong, VIC 3220, Australia

Manuscript received MonthTODO 19, 2020; revised Month 26, 2020.(PLACEHOLDER)

II. Related Works

A. Attacks in IoV

A survey of various attacks and detection mechanisms on VANETs and IoVs has been done by Sakiz et al. [2]. A brief description of the various attacks described is as follows.

- **Sybil Attack** A node pretends to have more then one identity.
- **DoS Attack** A Denial of Service or Distributed Denial of service attack aims to render the service unavailable by means of jamming, flooding etc.
- **Blackhole Attack** A node sends false routing information to make all other nodes try to route their packets through it, the packets are then dropped.
- **Wormhole Attack** Similar to black-hole attack, in a wormhole attack, two compromised nodes forward packets between each other after encapsulating them therefore the hop count is not affected. This makes these two nodes appear as the best route to send any packets.
- **Bogus Information Attack** A type of soft attack where a malicious node sends false information in the network.
- **Replay Attack** An attacker replays a message that was sent earlier out of context. Unlikely with the use of timestamps.

The bogus information attack mentioned above is based on the fact that vehicles in an IoV share information among each other and use that information in various protocols. In this attack a malicious node disseminates false information with the aim of manipulating the behaviour of other nodes, this effect is increased if the attacker is moving around swiftly[3]. Various variations of this attack have been studied by Sakiz et al.[2] such as False Position Information[4], Sensor Tampering, Illusion Attack[5] and GPS Spoofing/Tunnel Attack[6]. Various methods have been proposed for the detection of false messages as described in the following subsection (II-B). These can help a vehicle identify which messages to ignore and the results of these methods could further be used to detect and purge attackers from the network for which voting, evaluation and reputation based mechanisms have been suggested[7](*Local Eviction of Attackers by Voting Evaluators*)[8].

B. False Message Detection

Lo et al. [5] propose a mechanism called *Plausibility Validation Network (PVN)* which is capable of checking the data received from sensors, or other vehicles, and validating it. It includes a plausibility network and a rule database. For each category of messages the various rules are used to detect information that logically must be false based on known truths. Another message filtering model is proposed by Kim et al. [8]. The model includes a threshold curve and a certainty of event (CoE) curve. The CoE, which indicates the confidence level of a received message, is calculated by combining the data from various sources such as local sensors, RSUs and reputation mechanism. The solution relies on honest majority. The threshold curve shows the insensitivity of the driver with respect to the distance to the event. Sensitivity and the distance to the event are inversely proportional. Therefore, while the threshold value is decreasing, the CoE value keeps increasing and, if it exceeds the threshold value which is assigned according to the application, the driver is warned with an alert message. The paper suggested the use of a rudimentary reputation system as an input factor for the CoE score among five other factors.

C. Trust Management and Reputation

Iqbal et al. [9] enumerated various factors involved in the conceptualisation of a trust system in SIoV, which differs from IoV by the fact that objects in SIoV maintain social relationships. Trust management mechanism in SIoV serves the purpose of establishing a node as trustworthy and to gauge the trustworthiness of other nodes in the network. The eight main factors enumerated for evaluation of trust were:

1) Reputation - Score based on feedback.
2) Context
3) Environment
4) Goals
5) User expectations
6) Social relationships
7) Willingness to connect
8) Timeliness of evaluation

Of these eight factors, reputation plays a fundamental role. Various methods have been proposed of a reputation mechanism in VANETs. The utility of various reputation mechanisms proposed for MANETs is limited in VANETs due to the dynamic nature of the network where no two nodes are in range of each other for long periods. Various reputation mechanisms have been proposed for specific applications such as location verification[10]. The importance of a decentralised system for trust management in VANETs[11] has been recognised among other previous efforts surveyed by Soleymani et al.[12].

III. Proposed Methods

With the advent of 5G networks enabling rapid 2-way communication, we propose a primarily centralised system that leverages this connectivity for reputation score generation based on accuracy of information shared, where the information is not restricted to any specific type or application as long as it can be evaluated to be true or false by a node. The aim of this mechanism is to accurately predict the accuracy of information from sources based on feedback on that information and to filter out false feedback. This reputation mechanism may then be used as a sub module of a multi-factor trust management system. We describe our proposed system for reputation score generation in this section as well as propose methods of sharing feedback to the network and usage of the scores.

We have divided the section into three subsections. The first subsection details how the feedback is created, propagated and collected in the network, in the second sub-section, methods of score generation are described and the third sub-section details the proposed method for the distribution/application of the scores generated. It is assumed that the RSUs can communicate with each other and therefore can distribute the computation and maintain shared storage in the from of IPFS etc. However, the details of it are beyond the scope of this article. For simplicity the network of RSUs is referred to as a single unit henceforth in this section.

A. Feedback Generation and Collection

Nodes in an IoV may be able to classify some of the information they receive as true or false either by means of self observation or context aware deduction using primarily data centric methods of false information detection such as the those proposed by Kim et al.[8] and Lo et al.[5], as previously mentioned in Section II-B. If a message is determined to be true or false by a receiving node, it can share feedback on it with the other nodes of the network. The usage of this feedback is detailed in subsequent subsections. In the proposed system feedback is shared primarily in the form of a report comprising of 4 elements: the reportee's id (or address), the reporter's id (or address), the message id (or a unique message identifier), and a boolean value used to indicate if the message being reported on was found to be true or false by the reporter.

1) Feedback Generation by Nodes: The feedback exchange in the network is triggered by the following events:

- **Message Classified:** Whenever the truthfulness of a received message is determined, a report is created and shared on the network.
- **Message received from a sender for the first time:** In order to bring new nodes joining a network up to par with the rest of the network, whenever a node a receives a message from a node b on which it has no feedback information, it (a) sends a ”Request for Reports” message containing only b's id.
- **Request for Reports Received:** When a Request for Reports on any node b is received by some node a, it sends a ”Reports Dump” message containing all the reports it has sent on b that it contains in storage.

2) Feedback Collection at RSU: Three lists of messages are maintained: Current Scope, Staged Scope and Archived Scope. Two data objects, referred to as baskets, are used to store and process the reports received, all reports that are received are put into one of two baskets:

1) **Current Basket**: if the message that the report is on is not in Staged Scope or Archived Scope, the message is added to the Current Scope and the report is collected in the Current Basket.
2) **Staged Basket**: Only reports on messages that are listed in the staged scope are collected in the staged basket.

If a report is received on a message that is in the Archived Scope, it is ignored. Each basket stores the reports that are inserted in it and it's implementation can include variables to maintain some metadata on the reports to facilitate the score generation process which is detailed later. At regular intervals of time, which roughly equals the average amount of time between a node sending a message and another node receiving a report on that message, a ”Stage Shift” event is triggered. When Stage shift occurs the elements of the staged scope are added to archived scope. Secondary scores and fuzzy truth-values of messages are calculated based on the contents of the staged basket before it is emptied, this process is elaborated in following subsections. These truth-values are then inserted into logs, which is used to generate the primary scores. After this the contents of the Current Basket are shifted to the Staged Basket before it is emptied. Corresponding operations are performed on the staged and current scope lists. The logs for each vehicle contain the calculated truth-values of messages sent by it, and other counters that are used to calculate the primary scores for it with optimal complexity. It can be summarised that the archived scope is the list of messages for which the truth-values have been calculated and inserted into the logs, the staged scope is the list of messages for which the reports are still being collected and that will be processed and inserted into logs at the next stage shift event, and the current scope is the expanding list of messages that will form the fixed staged scope list at the next stage shift event.

3) RSU Results Broadcasting and incorporation at Nodes: As mentioned previously, the logs at the RSU contain the calculated truth-values of messages which are used to generate the Primary Scores, the calculation of these truth-values and Primary Scores is detailed in the subsequent subsection. In order to share it's results the RSU calculates the primary score of each node for various window sizes i.e [10,50,250,1250] or such and sends these (with other metadata) along with the list of blacklisted nodes in a message to all nodes in the network. Whenever a message from an RSU is received the node can make a copy of the blacklist to ignore reports from the blacklisted nodes, until the blacklist is updated again and it can insert dummy messages/reports from various nodes into it's own storage by reverse calculating from the received primary scores and metadata in order to emulate the same knowledge.

B. Score Generation

We describe the method of score-generation in this subsection. All data pertaining to reports and counters in this subsection and in Algorithms 1 to 6 refers to the data collected in the staged basket up until the stage shift event occurs and the various score generation methods are called, unless otherwise specified to refer to data in Logs. In order to generate an estimate of the overall accuracy of all the information shared by a node, the system calculates a truth-value for each message sent by that node based on all the feedback on that message, then the mean of the truth-values of a set of messages is taken. This estimate of the overall accuracy of a node is referred to as the Primary Score or the Reputation Score interchangeably in this article. Since the feedback may be deliberately falsified as well, it is first necessary to exclude the feedback from nodes that may be malicious. In order to distinguish nodes with false feedback another parameter is used which is referred to as the ”Secondary Score” for a node. The following sub-sub-sections describe the methods for ultimately generating the Primary Score for each node.

1) Secondary Score Generation: When ever a new report is received by the RSU, counters corresponding to the number

Algorithm 1 Calculate Median Implied Scores

INPUT:

1) N: Set of all nodes on which at least one report was received
2) $Count_{i,j}^{Total}$: Count of Total Reports on node i by node j $\forall i, j \in N$
3) $Count_{i,j}^{Positive}$: Count of Positive Reports on node i by node j $\forall i, j \in N$

OUTPUT: $Score_n^{MIS}$: Median Implied Score of node n $\forall n \in N$

for all $i \in N$ **do**
  $impliedScore_j \leftarrow Count_{i,j}^{Positive} / Count_{i,j}^{Total}$ $\forall j \in N - \{i\}$
  $Score_i^{MIS} \leftarrow$ median($impliedScore_j$ $\forall$ $j \in N - \{i\}$)
end for
return $Score_i^{MIS}$ $\forall i \in N$

Complexity: $\mathcal{O}(n^2)$.

Algorithm 2 Calculate Secondary Scores

INPUT:

1) N: Set of all nodes from whom at-least 1 report was received
2) $Count_{i,j}^{Total}$: Count of Total Reports on node n by node j $\forall i, j \in N$
3) $Count_{i,j}^{Positive}$: Count of Positive Reports on node n by node j $\forall i, j \in N$
4) $Score_n^{MI}$: MI Score of node n $\forall n \in N$

OUTPUT: $Score_n^{Secondary}$: Secondary Score of node n $\forall n \in N$

for all $i \in N$ **do**
  $TSE \leftarrow 0$
  $TotalReports \leftarrow 0$
  for all $j \in N \mid j \neq i$ **do**
    $impliedScore \leftarrow Count_{j,i}^{Positive} / Count_{j,i}^{Total}$
    $error \leftarrow Score_j^{MI} - impliedScore$
    $TSE \leftarrow TSE + (error^2 \times Count_{j,i}^{Total})$
    $TotalReports \leftarrow TotalReports + Count_{j,i}^{Total}$
  end for
  $Score_i^{Secondary} \leftarrow \frac{TSE}{TotalReports}$
end for
return $Score_i^{Secondary}$ $\forall i \in N$

Complexity: $\mathcal{O}(n^2)$.

Algorithm 3 Generate Blacklist

INPUT:

1) N: Set of all nodes for which a secondary score exists
2) $Score_n^{Secondary}$: Secondary Score of node n $\forall$ $n \in N$

OUTPUT: $Blacklist$: Set of all blacklisted nodes

$m \leftarrow$ median($\{Score_i^{Secondary} \mid i \in N\}$)
$ADm \leftarrow \emptyset$
for all $i \in N$ **do**
  $ADm \leftarrow ADm \cup \{|Score_i^{Secondary} - m|\}$
end for
$MADm \leftarrow$ median(ADm)
$threshold \leftarrow m + 2 \times MADm$
$Blacklist \leftarrow \emptyset$
for all $i \in N$ **do**
  if $Score_i^{Secondary} > threshold$ **then**
    $Blacklist \leftarrow Blacklist \cup \{i\}$
  end if
end for
return $Blacklist$

Complexity: $\mathcal{O}(n)$.

of positive and negative reports on a node and the number of positive and negative reports from a node on another node can be incremented. These counters can then be utilised to quickly calculate each node's median implied reputation score (MI Score), which is defined as the median of all the implied reputation scores for that node, where an implied reputation score of a node is the fraction of positive reports to total reports by another node. See Alg. 1. By assuming the MI score for each node as a benchmark, the deviation of the scores implied by a node's reports from the respective MI scores is utilised as a measure of the utility of it's reports. This deviation is measured in terms of Mean Squared Deviation per report and is termed as the Secondary Score of the Node. It is calculated for each node as described in Algorithm 2.

2) Blacklist Generation: The Secondary Scores are a measure of abnormality on the reports of a node. This measure may be used to scale down the effect of it's reports during the Primary Score generation, or more simply it may be used to blacklist nodes whose Secondary Score crosses a certain threshold. The latter approach was pursued by us during experimentation where a simple heuristic method of determining a dynamic threshold based on the distribution of the secondary scores was adopted. With the assumption of honest majority the median of secondary scores must not be more than that of a dishonest node with abnormally lower utility of reports. Therefore the threshold value was defined as a function of the median of all Secondary Scores and the median absolute deviation around that median. The process of the generation of the blacklist is as described in Algorithm 3

3) Primary Score Generation: The counters utilised during secondary score calculation may also be used to calculate estimates of accuracy for each node in constant time as described in Algorithm 4. This estimation is referred to as RAW Score and serves as a baseline during experiments, in itself the RAW score is highly inaccurate however it is quick to calculate therefore it may be employed at nodes where the usual method may not be applicable, this is detailed in Section III-C3.

With all the reports on each message indicating it to be either true or false, the mean of these reports can be calculated and treated as a fuzzy truth-value of it. As mentioned earlier

Algorithm 4 Calculate RAW Scores

INPUT:

1) N: Set of all nodes on whom atleast 1 report exists
2) $Count_i^{Total}$: Count of Total Reports on node i $\forall i \in N$ in Staged Basket as well as Logs.
3) $Count_i^{Positive}$: Count of Positive Reports on node i $\forall i \in N$ in Staged Basket as well as Logs.

OUTPUT: $Score_n^{RAW}$: RAW Score of node n $\forall n \in N$

```
for all i ∈ N do
    Score_i^RAW ← (Count_i^Positive / Count_i^Total)
end for
return Score_i^RAW ∀i ∈ N
```

Complexity: $\mathcal{O}(n)$.

the primary score is an estimation of the accuracy of all the information shared by a node, this can be trivially calculated as the mean of the calculated truth-value of a set of messages by that node. During this calculation the reports by blacklisted nodes are ignored. The process of calculating these truth-values is as described in Algorithm 5. These calculated truth-values for messages are then inserted into the logs. From these logs we generate the Primary Score which is the mean of the truth-values of a set of messages. Calculating the Primary Score over the all the messages would give a close estimate of a nodes actual accuracy assuming it remains consistent however it is not likely to reflect a change in the accuracy of a node quickly. On the other hand the Primary Score of a node over a small window of message truth-values is likely to fluctuate more on every stage shift and not be an accurate estimate of the behaviour of a node however it would reflect a sudden change in th accuracy of a node quickly. The process of generating Primary Scores over a given window size for all nodes is described in Algorithm6

C. Usage

Trends in the Primary Scores of each nodes, over different window sizes, can be tracked to detect malicious behaviour such as perpetual inconsistency, drastic changes etc. besides the trivial case of being low. Likewise Secondary Scores can be monitored as well to identify and remove nodes from the network. In the process of calculating the MI Scores prior to Secondary Score Calculation (See Algorithm 1) the distribution of the implied scores can be analysed by using statistical methods of multimodality detection such as Dip test[13] or Silverman's test[14], where a high degree of bi-modality can be an indicator of collusion to manipulate the reputation system. These options of deeper analysis have not been further explored in this article however have been mentioned here as an indicator of possible extension of the methods. In the scope of this article the calculated truth values are used to calculate the Primary Scores at various sizes and share them with the Nodes where their utilisation can be done in a number of alternative ways. The application

Algorithm 5 Calculate Truth-values

INPUT:

1) N: Set of all nodes
2) M_i: Set of all messages (ids) from node i $\forall$ $i \in N$
3) $Report_j^{i,m} \in \{0,1\}$: Alleged Validity of message m by node i in a Report by node j $\forall m \in M_i | Report_j^{i,m}$exists $\forall$ $i, j \in N | i \neq j$
4) B: Set of all blacklisted nodes

OUTPUT:

1) M'_i: Set of all messages (ids) from node i for which a truth-value is calculated $\forall$ $i \in N$
2) tv_m^n: Truth-value of message m from node n $\forall m \in M'_n$ $\forall n \in N$

```
for all n ∈ N do
    M'_n ← ∅
    for all m ∈ M_n do
        R ← {Report_j^{n,m} | j ∈ N−B and Report_j^{n,m} exists}
        if |R| ≠ 0 then
            tv_m^n ← (Σ_{r∈R} r) / |R|
            M'_n ← M'_n ∪ m
        end if
    end for
end for
return tv_m^n ∀m ∈ M'_n ∀n ∈ N
```

Complexity: $\mathcal{O}(n^2 m)$.

Algorithm 6 Calculate Primary Scores

INPUT:

1) N: Set of all nodes in Logs
2) M_i: List of all messages (ids) in Logs from node i in reverse order of arrival $\forall$ $i \in N$
3) tv_m^n: Truth-value of message m from node n $\forall m \in M_n$ $\forall n \in N$
4) w: Size of the window for which average truth-value needs to be calculated

OUTPUT: $Score_n^{Primary,w}$: Primary Score of Node n calculated over a window of size w $\forall n \in N$

```
for all n ∈ N do
    M' ← M_n[1 to w]
    sum ← Σ_{m∈M'} tv_m^n
    Score_n^{Primary,w} ← sum/w
end for
return Score_n^{Primary,w} ∀n ∈ N
```

Complexity: $\mathcal{O}(nm)$.
Complexity *for optimized implementation of logs*: $\mathcal{O}(n)$.

of the generated scores could be as an input to the false message detection system that generated the feedback this system depends on, as some methods consider reputation of the sender as a parameter for evaluation[8]. The reputation score could also used to select vehicles that must be removed from the network or it could be used as a factor from a broader trust management system[9].

1) Usage at RSU: As previously mentioned in Section III-A, at the RSU the Primary Scores are calculated at certain window sizes for all nodes and this data along with some meta data such as the id or timestamp of the last message considered from each node is sent along with the blacklist as a message to the network. At the nodes this data of primary scores along with the metadata is used to create dummy message truth-values or reports (depending on the usage) to emulate the data that would produce these Primary Scores, This is done to reduce the amount of information that needs to be sent over the network. In a fast 5G network, all the truth values in a certain window may be shared for each vehicle.

2) Usage at Nodes - Disabled Node: The nodes can depend on the RSU entirely to provide the Primary Scores. Pessimistically minimum of the primary scores of a node (over different window sizes) can be taken.

Advantages: Low Computation at nodes

Disadvantages:

- Less adaptable to change in behaviour of nodes, scores remain static between RSU Stage Shifts
- System is Entirely dependant on RSU

3) Usage at Nodes - Adapted RAW Score Algorithm: Reports can be used by nodes to calculate Reputation Scores by using an adaptation of the RAW Score Algorithm at the RSU (See Alg. 4) where the average of the last w' reports on a node is taken as a Primary score over a window size of w'. When a RSU Broadcast is received, for the smallest window size w_1, w_1 messages with truth-value equal to $Score^{Primary,w_1}$ are emulated. For each subsequent window size w_i for which a primary score exists on nodes, $w_i - w_{i-1}$ messages with truth-value calculated based on 1 are emulated. For each message to be emulated x number of dummy reports, whose average approximately equals the desired truth-value, are inserted into the logs, where x is the average number of reports per message (this can be included as part of the metadata in the RSU broadcast). The window sizes to be used when calculating Primary Scores locally can be calculated by scaling the window sizes used at the RSU by a factor of x.

$$tv = \frac{Score^{Primary,w_i} \times w_i - Score^{Primary,w_{i-1}} \times w_{i-1}}{w_i - w_{i-1}} \tag{1}$$

Advantages:

- More adaptable to change in behaviour of nodes, scores are dynamic between RSU Stage Shifts
- Moderate Computation at nodes

Disadvantages:

- System is largely dependant on RSU, Estimates will become highly inaccurate over time without RSU updates

4) Usage at Nodes - Adapted Primary Score Algorithm: Reports can be used by nodes to calculate Reputation Scores by using an adaptation of the Primary Score Algorithm at the RSU (See Alg. 5 and 6) where the average of the truth-values of the last w messages on a node is taken as a Primary score over a window size of w. Messages with less than a certain number of reports on it can be kept in a buffer while more reports are received. Whenever a new report on a message is received it's truth-value is accordingly updated and so is the Primary Score over windows that includes the message. When a RSU Broadcast is receivedf, for the smallest window size w_1, w_1 messages with truth-value equal to $Score^{Primary,w_1}$ are inserted. For each subsequent window size w_i for which a primary score exists on nodes, $w_i - w_{i-1}$ messages with truth-value calculated as shown in 1 are inserted. Reports from nodes that are blacklisted are purged and so are the reports on messages that have been evaluated by the RSU for Score generation. This data can be provided in the metadata of the RSU Broadcast.

Advantages:

- More adaptable to change in behaviour of nodes than disabled node, scores are dynamic between RSU Stage Shifts
- System is Less dependant on RSU, Estimates will remain accurate to a high degree over time without RSU updates in the absence of malicious nodes that share false feedback.

Disadvantages:

- System would be adversely affected in the absence of RSU if malaicious nodes send false feedback.
- Higher amount of computation and storage at node

IV. COMPARISON

Various Trust systems have been categorised and compared by Indu et al. [15]. Based on the factors used in the aforementioned paper for classification, we compare our system in a qualitative manner with two other recent trust systems for IoVs. These systems are "Trust Management System for SIoV" by Fangyu Gai et al.[16] and "Trust Management Scheme with Affinity Propagation" by Shu Yang et al.[17], the former is referred to as the ratee based system and the latter is referred to as the affinity propagation model henceforth in this section.

A. Centralised vs. Self-Organised/Distributed Trust

Centralised Trust systems require a central body for calculation of trust. This can be a advantageous as higher computational and storage capabilities of RSUs can be leveraged, however, it also creates a single point of failure. The ratee based system has the aforementioned drawback due to a centralised system, whereas the affinity propagation model does not. Our system is primarily centralised however it can function even in the absence of the RSUs as explained in section III-C4, therefore it will not fail completely in case there is a failure at the RSU.

B. Data-Based vs. Entity-Based

In Entity-Based systems the trust is assigned to entities in the network i.e nodes/vehicles, as the name suggests, this can be useful for detection of malicious actors that can be evicted/ignored. In Data-Based systems the trust is assigned to the data received, these are most often distributed, quick and robust. In the comparison done by Indu et al.[15] it was found that a majority of trust systems for IoV and VANETs are Data-Based. The affinity propagation model is Data-Based whereas the ratee based system is Entity-Based. It is speculated by Indu et al.[15] that a combined system would be best suited for IoVs. Although the proposed system is entirely Entity-Based, it can be layered on top of a distributed and entirely Data-Based system where the trust score of the data recieved can be treated as the ”truthfulness” of it, some thresholding can be applied to convert the trust score to true or false. This can be done to create a combined trust system from a Data-Based system thereby giving the advantages of having trust assigned to entities.

C. Rate of Trust Propagation

An important yardstick to evaluate a trust/reputation system is the rate of propagation. This can be viewed in two ways, one is how much time does it take for the reputation of a node to be affected according any other node after it starts to display anomalous behaviour and the second is how much does it take for the change reputation of some node to propagate across the network. In the experimental analysis done by Indu et al.[15] it was found that the rate of Propagation of Trust was slightly better in the affinity propagation model. In the ratee based model it was slower which can be attributed to the centralised system in it. In the system proposed in this article the rate of propagation is expected to be high due to two reasons, firstly, although reputation scores are broadcasted by a central unit, between broadcasts nodes can change the reputation score of vehicles according to them based on what they observe and also based on the constant feedback received from the network which ensures rapid propagation of information and, secondly, since the score is calculated for each node based on its previous messages in various window sizes, as explained in section III-C, the shortest sized window will be very sensitive to recent changes in behaviour and thus any anomalous behaviour would be detected before more than a few false messages are sent.

V. EXPERIMENTS

The core system was tested by software simulation in various scenarios. Details on the simulation set-up and parameters can be found in Appendix A. Three different cases in two separate environments were simulated to give a total of six scenarios. The Environments were as follows:

A. Environments

1) City: The road network of Melbourne CBD was taken with vehicles spawning at random locations, at random times, driving to a random destinations, then going off-line. Lifespans of individual vehicles were short, however the network is always densely populated with new vehicles appearing as old ones go offline. Meant to simulate the situation of traffic in a busy city centre. See Fig.. 1

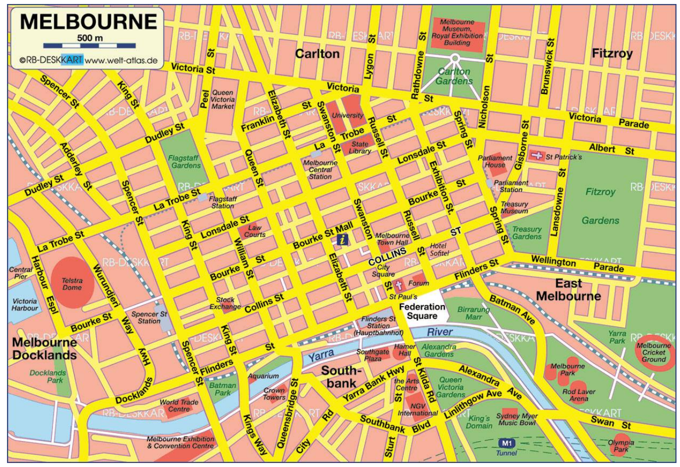


Fig. 1: Melbourne CBD Road Network

2) Highway: A flow of a fixed number of vehicles was simulated to run on a fixed route, mostly comprised of straight multi-lane roads. All nodes had larger lifespans than before. By means of overtaking and other reasons, the order of initialisation was not maintained as the order in which the nodes continued to travel. It was meant to simulate the situation of vehicles moving on a highway.

B. Situations

Different situations were simulated to analyse the system's performance against various attack vectors of compromising the the systems ability to produce accurate estimates. The behaviour of a regular node was programmed as follows: Overall accuracy of Sent Messages is around 90%. The node can determine the truthfulness of 60% messages it receives with an accuracy of 95%. Further details of a regular node's behaviour can be found in appendix A.

1) Situation 0: The first situation considered was that of a few (10%) malicious nodes that send false information on the network with an accuracy of 5%. The rest of the parameters controlling their behaviour was the same as regular nodes. This situation enables us to analyse the system's performance at detecting false messages and malicious nodes in a scenario where no node is sharing falsified feedback to jeopardise the reputation system. This can then be compared with other situations to spot degradation in performance.

2) Situation 1: In the second situation, over and above the factors present in situation 0, 10% Nodes were programmed to send reports with an accuracy of 5% i.e 95% reports generated by them are falsified and they send reports on 100% messages they receive as opposed to 60%, this was implemented to observe the degradation in the accuracy of estimates with false reports being shared by nodes and the ability of the system to detect the nodes sending falsified reports.

3) Situation 2: In the third and final situation, over and above the factors present in situation 0, 20% of nodes were programmed to target 5% of nodes specifically. These colluding nodes would send falsified feedback on every message received from target nodes while sending useful reports otherwise. This was implemented to analyse the systems ability

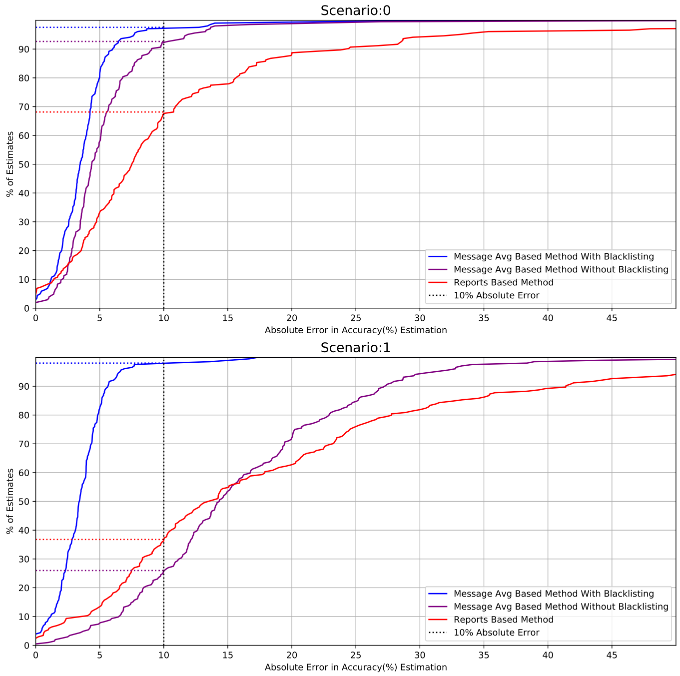


Fig. 2: Absolute error (difference between actual accuracy of node and estimated accuracy) vs. Percentage of nodes for which there was less than 'x'% error in Scenario 0 (above) and Scenario 1 (below)

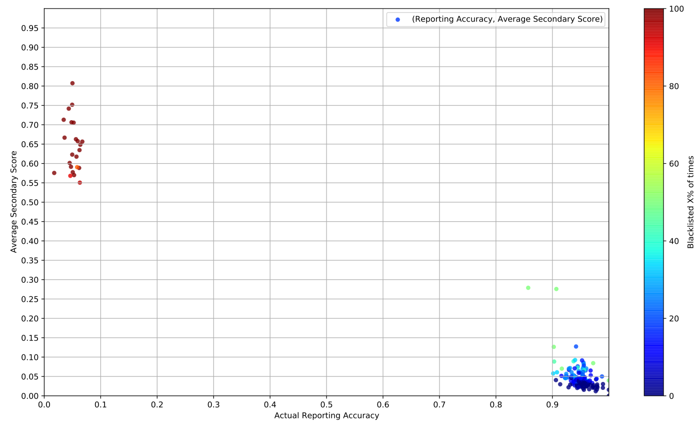


Fig. 3: Mean Secondary Score vs. Accuracy of Reports Sent, Scatter plot for Scenario 1

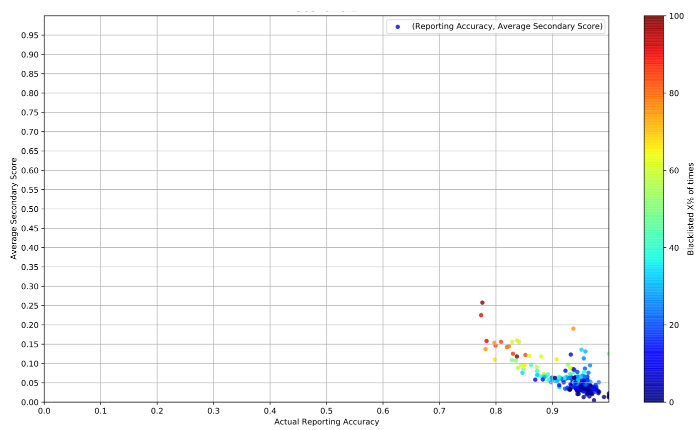


Fig. 4: Mean Secondary Score vs. Accuracy of Reports Sent, Scatter plot for Scenario 2

to detect convoluted methods of compromising it's ability of estimating accuracy where a large percent of malicious nodes send mostly useful feedback in order to not get blacklisted but target specific nodes, either regular or malicious, to alter their primary score.

C. Scenarios

The various permutations of environments and situations were termed as follows.

- Scenario 0: Situation 0 in City
- Scenario 1: Situation 1 in City
- Scenario 2: Situation 2 in City
- Scenario 10: Situation 0 on Highway
- Scenario 11: Situation 1 on Highway
- Scenario 12: Situation 2 on Highway

VI. Results

Various plots on the results of the simulations for each scenario are presented in Appendix. B. It can be observed from the results that in the absence of falsified reports, the system can estimate the accuracy of almost all (97%) nodes in city environment and all nodes in highway environment with less than 10 absolute error where accuracy and scores were measured in percentage.

Comparing the system's performance between scenario 0 and scenario 1 (where falsified reports were being sent) shows that significant percentage of scores are highly inaccurate for scores calculated without blacklisting. In scenario 0, 92.2% of accuracy estimates without blacklisting had less then 10 absolute error (0.1 when normalised) where accuracy and scores were measured in percentage. In scenario 1 this dropped down to 25.5% of estimates. Whereas the same measure for score calculation with blacklisting remained about the same. This can be seen in Fig. 2. Similar observations can be made when comparing scenario 10 and 11. This lack of degradation of the system can be attributed to it's ability to accurately identify nodes that send false feedback all the time. This can be seen in Fig. 3, where nodes with low utility of reports were almost always blacklisted as indicated by the deep red colour. In Scenarios 2 and 12, where a more intelligent approach to sabotage the system's accuracy estimate (score) of a node was made by 20% nodes in the network that acted like regular nodes except when sending reports on 5% nodes that were the targets for the collusion, the accuracy of those target nodes was highly inaccurate as expected, around 30% absolute deviation from the actual value. With blacklisting, however, the errors were minimal , around 6% absolute deviation from the actual value. This lack of error can again be attributed to the system's ability to accurately identify and blacklist nodes that send false feedback even some of the times. This can be observed from in Fig. 4 where nodes with slightly lower utility of reports blacklisted sometimes to most often as indicated by the light blue to red colours. It is important to note that the special collusion detection methods of multi-modality detection[13][14] as explained in section III-C was not leveraged during experiments.

VII. Conclusion

In this paper we proposed a primarily centralised reputation mechanism that relies on feedback generated by nodes using false information detection mechanisms to first evaluate the truth value for each message, after filtering out the false/malicious feedback using rudimentary statistical methods. Then, using these truth values, for messages the reputation score was calculated. We further proposed possible extensions of this method such as specialised method for detection of collusion, and various ways in which this data can be shared by the RSUs to the rest of the network. We also proposed the ways in which the nodes can utilise this data to generate reputation scores for nodes at their end and to dynamically alter this score in between RSU broadcasts. The core method for reputation score calculation was tested using software simulation where it was observed that the system was highly accurate at estimating the accuracy of nodes (reputation) when the feedback was entirely genuine. The estimates naturally became much more inaccurate with falsified feedback from ten percent of the network however this change in error was almost entirely eliminated with the proposed method for false feedback filtration. The same observations were made in a scenario where a large group of attackers maintained a good ratio of useful to false feedback by colluding to targeting specific few nodes.

Appendix A
Simulation Set-Up

A. *Traffic Simulation*

Traffic was simulated on SUMO. For city environment the road network of Melbourne CBD was generated using the osm web wizard script. Vehicle trips for this road network were created by a call to the random trips script from within the osm web wizard script. Routes for the generated random passenger trips were calculated with explicitly calling the duarouter method of SUMO. For the Highway Scenario, road network of larger Melbourne area was generated using the osm web wizard script. After this two points on the network, one in front of Deakin University, Burwood, and one in Fitzroy were chosen as starting and ending points. A route for the same was calculated using Duarouter and a flow of 50/100 vehicles was generated on that route.

B. *Network/Application Simulation*

Veins on OMNET++ was utilised to simulate the network and the application. The default parameters in veins for antenna strength were utilised which are based on [18], same applies to 11p parameters and so on.

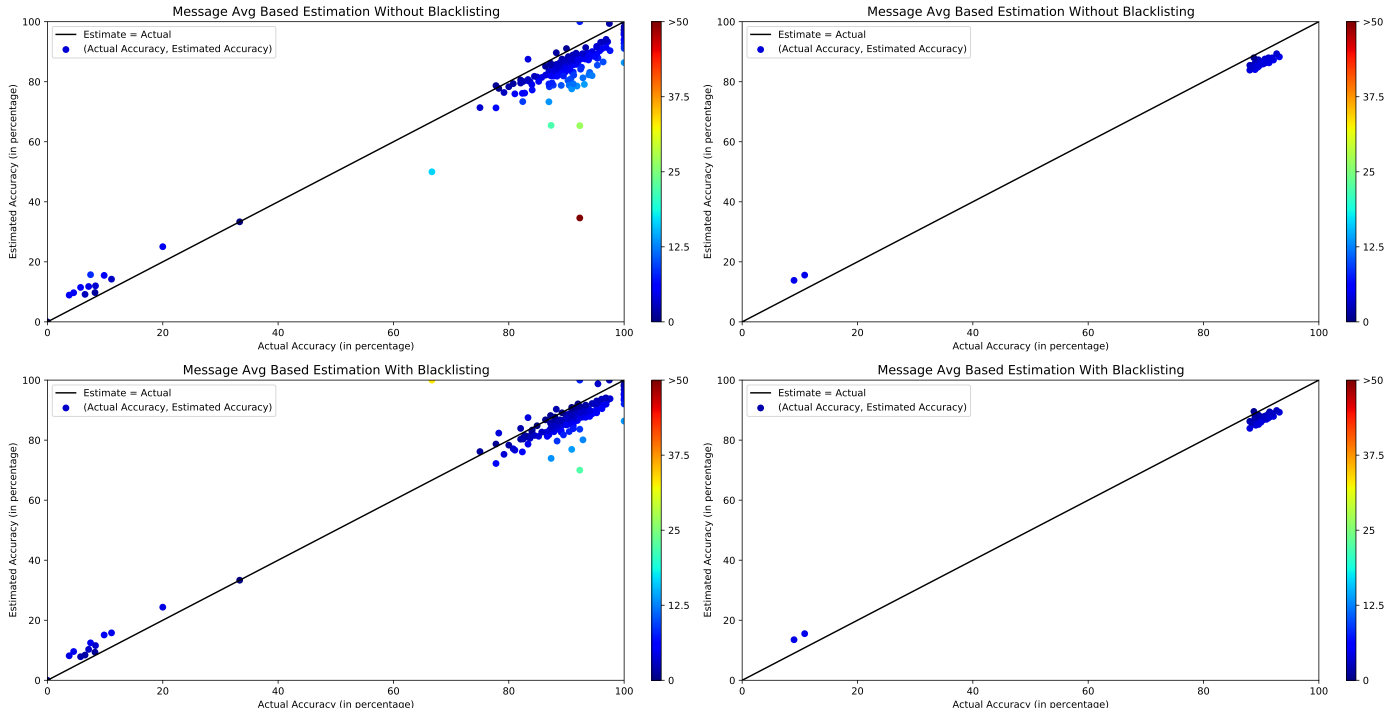
C. *Regular Nodes*

- Overall accuracy of Sent Messages is around 90%.
- Can determine the truthfulness of 60% messages it receives with an accuracy of 95%.
- Sends a message on a network every 4s$\pm$ 2000ms.
- Sends a report (its evaluation of a received message, if evaluated) in 2s$\pm$ 1000ms
- Sends a request on the network for all nodes to send their evaluations of all messages from a node when a message from a node is encountered for the first time in 1s$\pm$ 500ms
- Can oblige such a request in 1s$\pm$ 500ms

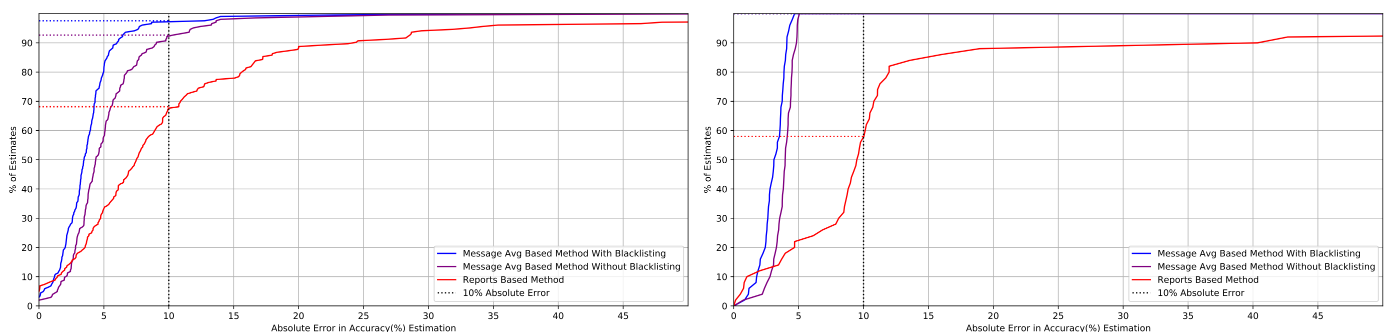
Other nodes' behaviour was different from a regular in only ways previously explained in Section V (V-B2 and V-B3).

Appendix B
Simulation Results

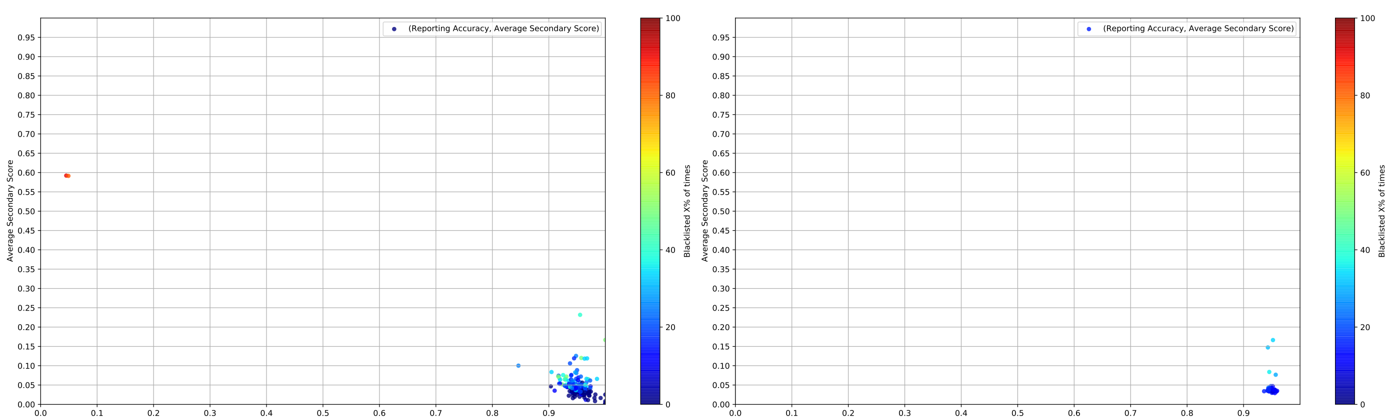
The the performance of the system in Situations 0 (Scenario 0 & 10), 1 (Scenario 1 & 11) and 2 (Scenario 2 & 12) is visualised in Figures B.1, B.2 and B.3 respectively. In these figures "Message Average Based Estimation with Blacklisting" refers to the proposed system, "Message Average Based Estimation without Blacklisting" refers to score calculation without blacklisting and "Reports Based Estimation" refers to the RAW Score (See Section III-B)



(a)

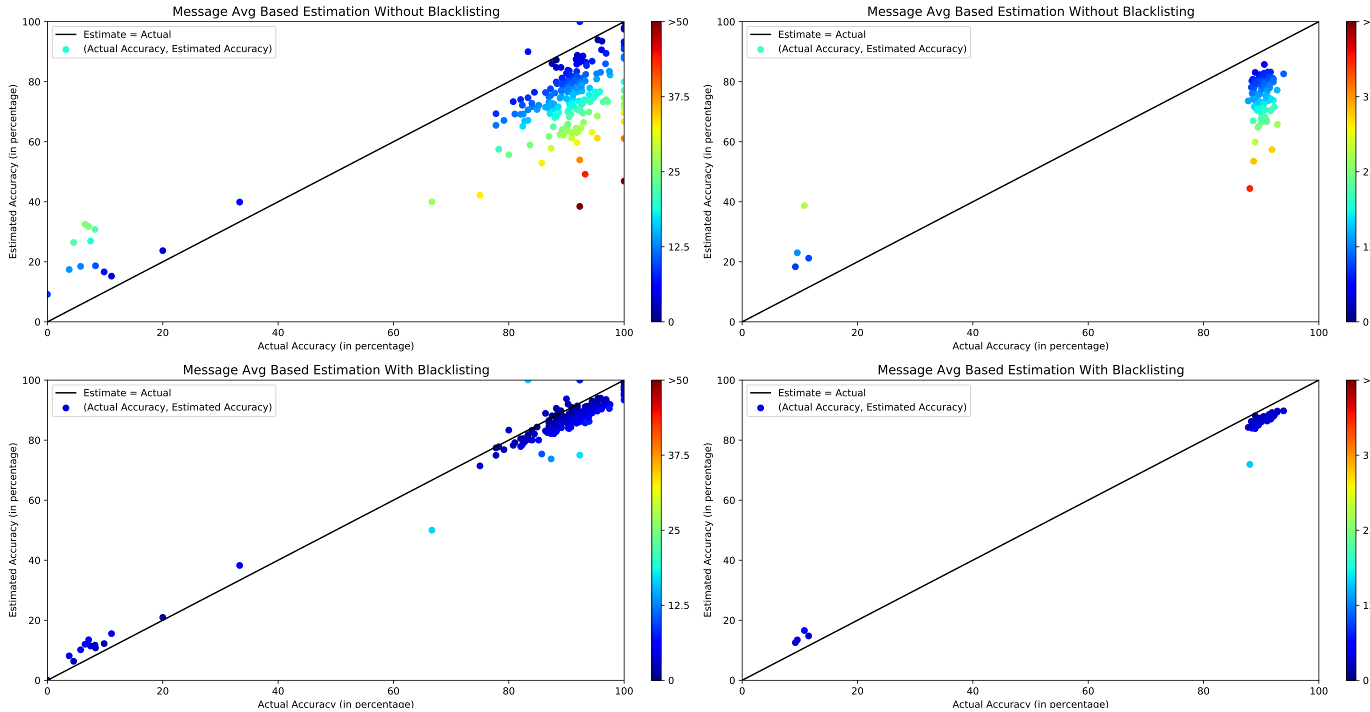


(b)

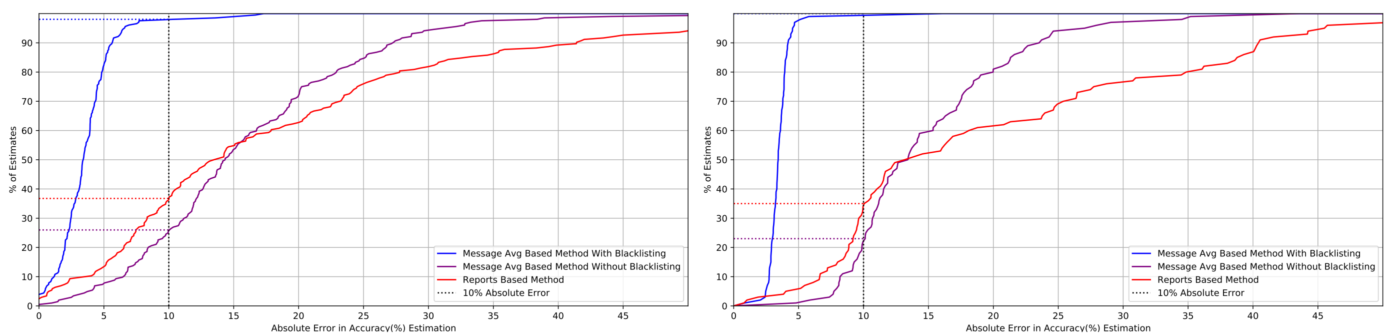


(c)

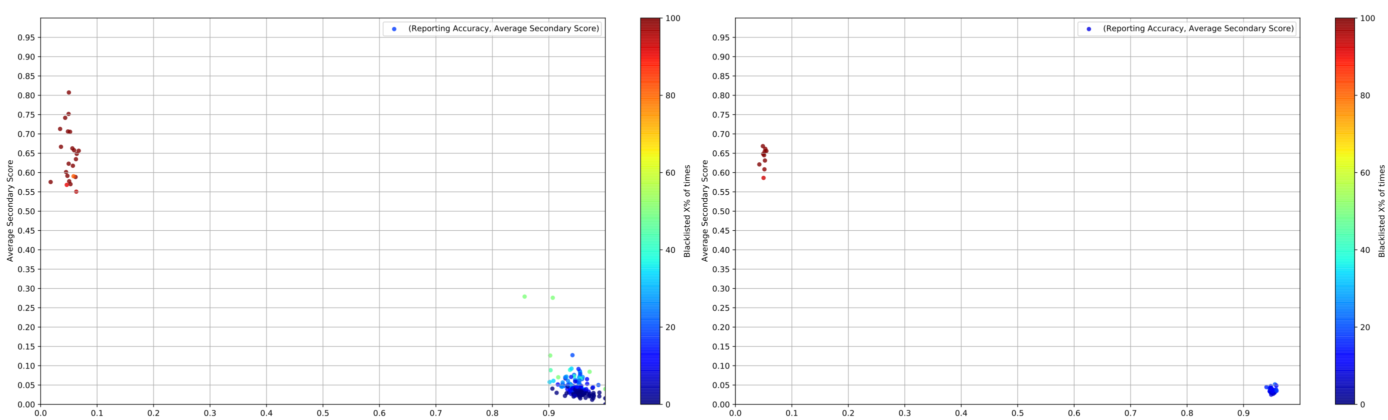
Fig. B.1: Graphs for Results of Scenario 0 and 10. (a) Primary Score vs. Actual Accuracy - Scatter plot, without (above) and with (below) blacklisting for Scenario 0 (left) and Scenario 10 (right) (b) Absolute error (difference between actual accuracy of node and estimated accuracy) vs. Percentage of nodes for which there was less than 'x'% error for Scenario 0 (left) and Scenario 10 (right) (c) Mean Secondary Score vs. Accuracy of Reports Sent - Scatter plot for Scenario 0 (left) and Scenario 10 (right)



(a)

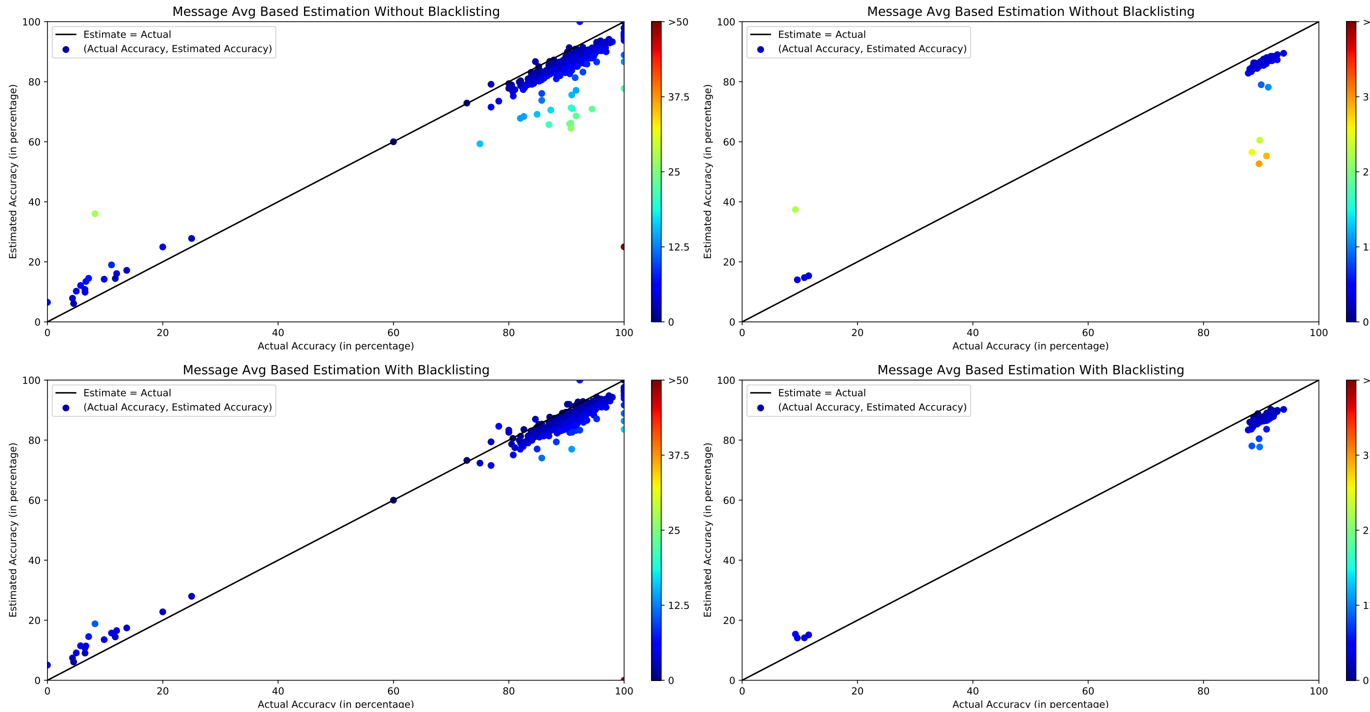


(b)

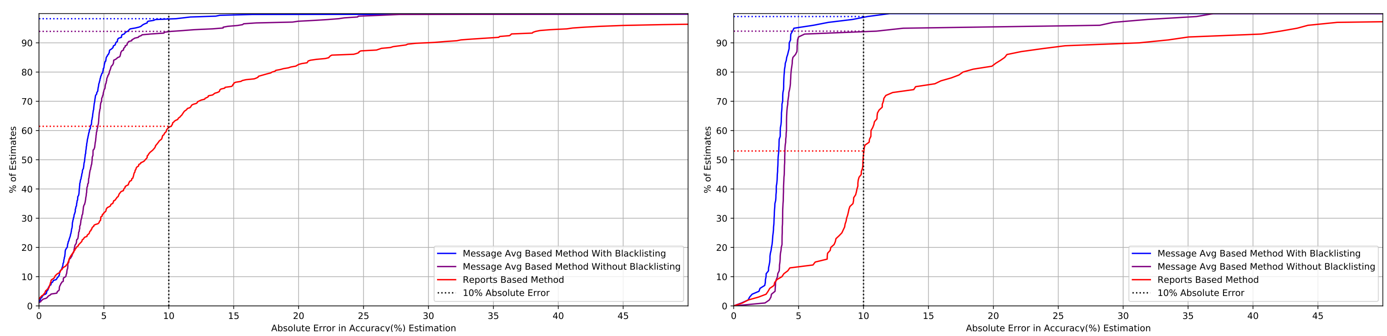


(c)

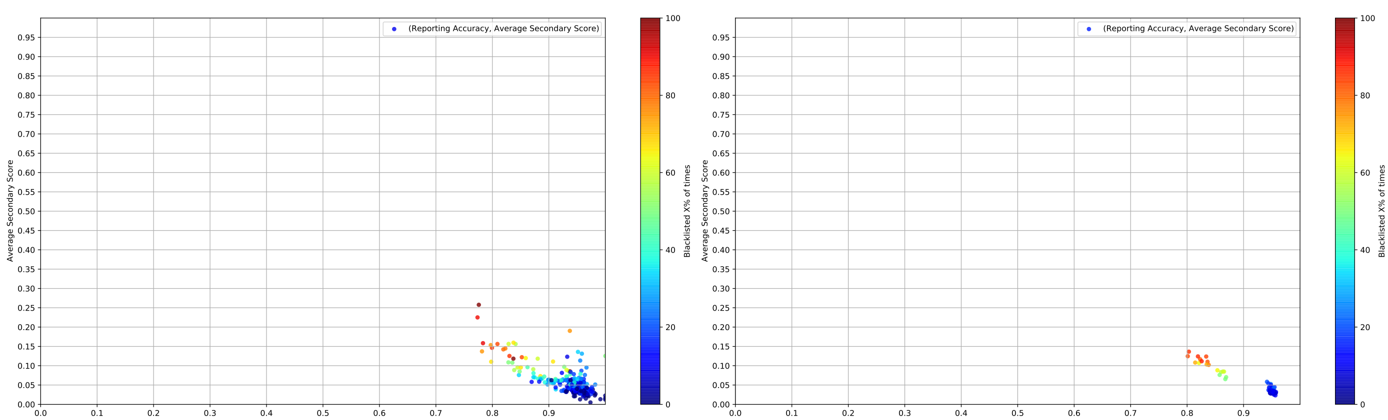
Fig. B.2: Graphs for Results of Scenario 1 and 11. (a) Primary Score vs. Actual Accuracy - Scatter plot, without (above) and with (below) blacklisting for Scenario 1 (left) and Scenario 11 (right) (b) Absolute error (difference between actual accuracy of node and estimated accuracy) vs. Percentage of nodes for which there was less than 'x'% error for Scenario 1 (left) and Scenario 11 (right) (c) Mean Secondary Score vs. Accuracy of Reports Sent - Scatter plot for Scenario 1 (left) and Scenario 11 (right)



(a)



(b)



(c)

Fig. B.3: Graphs for Results of Scenario 2 and 12. (a) Primary Score vs. Actual Accuracy - Scatter plot, without (above) and with (below) blacklisting for Scenario 2 (left) and Scenario 12 (right) (b) Absolute error (difference between actual accuracy of node and estimated accuracy) vs. Percentage of nodes for which there was less than 'x'% error for Scenario 2 (left) and Scenario 12 (right) (c) Mean Secondary Score vs. Accuracy of Reports Sent - Scatter plot for Scenario 2 (left) and Scenario 12 (right)

REFERENCES

[1] C. de Looper, "What is 5g? the next-generation network fully explained — digital trends," 2020. [Online]. Available: https://www.digitaltrends.com/mobile/what-is-5g/

[2] F. Sakiz and S. Sen, "A survey of attacks and detection mechanisms on intelligent transportation systems: Vanets and iov," *Ad Hoc Networks*, vol. 61, 03 2017.

[3] T. Moore, M. Raya, J. Clulow, P. Papadimitratos, R. Anderson, and J. Hubaux, "Fast exclusion of errant devices from vehicular networks," in *2008 5th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, 2008, pp. 135–143.

[4] T. Leinmller and E. Schoch, "Greedy routing in highway scenarios: The impact of position faking nodes," in *Workshop On Intelligent Transportation (WIT 2006)*, 01 2006.

[5] N. Lo and H. Tsai, "Illusion attack on vanet applications - a message plausibility problem," in *2007 IEEE Globecom Workshops*, 2007, pp. 1–8.

[6] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *J. Comput. Secur.*, vol. 15, no. 1, p. 3968, Jan. 2007.

[7] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, and J. Hubaux, "Eviction of misbehaving and faulty nodes in vehicular networks," *IEEE Journal on Selected Areas in Communications (WIT 2006)*, vol. 25, no. 8, pp. 1557–1568, 2007.

[8] T. H.-J. Kim, A. Studer, R. Dubey, X. Zhang, A. Perrig, F. Bai, B. Bellur, and A. Iyer, "Vanet alert endorsement using multi-source filters," in *Proceedings of the Seventh ACM International Workshop on VehiculAr InterNETworking*, ser. VANET 10. New York, NY, USA: Association for Computing Machinery, 2010, p. 5160. [Online]. Available: https://doi.org/10.1145/1860058.1860067

[9] R. Iqbal, T. A. Butt, M. Afzaal, and K. Salah, "Trust management in social internet of vehicles: Factors, challenges, blockchain, and fog solutions," *International Journal of Distributed Sensor Networks*, vol. 15, no. 1, p. 1550147719825820, 2019. [Online]. Available: https://doi.org/10.1177/1550147719825820

[10] M. Fogue, F. J. Martinez, P. Garrido, M. Fiore, C. Chiasserini, C. Casetti, J. Cano, C. T. Calafate, and P. Manzoni, "Securing warning message dissemination in vanets using cooperative neighbor position verification," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 6, pp. 2538–2550, 2015.

[11] Z. Huang, S. Ruj, M. Cavenaghi, and M. Stojmenovic, "A social network approach to trust management in vanets," *Peer-to-Peer Networking and Applications*, vol. 7, 09 2014.

[12] S. Soleymani, H. Abdullah, W. Hassan, H. Anisi, S. Goudarzi, M. Baee, and M. Satria, "Trust management in vehicular ad hoc network: a systematic review," *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, 05 2015.

[13] J. A. Hartigan and P. M. Hartigan, "The dip test of unimodality," *The Annals of Statistics*, vol. 13, no. 1, pp. 70–84, 1985. [Online]. Available: http://www.jstor.org/stable/2241144

[14] B. W. Silverman, "Using kernel density estimates to investigate multimodality," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 43, no. 1, pp. 97–99, 1981. [Online]. Available: https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.2517-6161.1981.tb01155.x

[15] Indu and S. Khara, "Internet of vehicles (iov): Evolution, architectures, security issues and trust aspects," *International Journal of Recent Technology and Engineering*, vol. 7, Mar. 2019. [Online]. Available: https://www.ijrte.org/wp-content/uploads/papers/v7i6/E2106017519.pdf

[16] F. Gai, J. Zhang, P. Zhu, and X. Jiang, "Trust on the ratee: A trust management system for social internet of vehicles," *Wireless Communications and Mobile Computing*, vol. 2017, pp. 1–11, 12 2017.

[17] S. Yang, Z. Liu, J. Li, S. Wang, and F. Yang, "Anomaly detection for internet of vehicles: A trust management scheme with affinity propagation," *Mobile Information Systems*, vol. 2016, pp. 1–10, 03 2016.

[18] D. Kornek, M. Schack, E. Slottke, O. Klemp, I. Rolfes, and T. Krner, "Effects of antenna characteristics and placements on a vehicle-to-vehicle channel scenario," in *2010 IEEE International Conference on Communications Workshops*, 2010, pp. 1–5.

Rohan Dahiya Rohan Dahiya was born in New Delhi, India in 1997. He is currently pursuing the B.Tech. degree in computer science and engineering with specialisation in information security from Vellore Institute of Technology, TN, India.
He has worked as an Intern at Volon Cyber Security in the summer of 2018 and at GAP IT Services India in the summer of 2019. He is was a visiting researcher at Deakin University, Burwood, VIC, Australia from December 2019 till May 2020.

Frank Jiang Frank Jiang received the masters degree in computer science from the University of New South Wales (UNSW), Australia and the Ph.D. degree from The University of Technology Sydney, Australia. He gained three and a half years of postdoctoral research experience at UNSW. He is currently a Senior Lecturer of cyber security with the School of Info Technology Campus, Deakin University, Australia. He has published over 100 highly reputed SCI/EI indexed journal/conferences papers. His main research interests include data-driven cyber security, predictive analytics, biologically inspired learning mechanism, and its application in the complex information security systems.

Robin Doss Robin Doss (Senior Member, IEEE) received the B.E. degree in electronics and communication engineering from the University of Madras, India, and the masters and Ph.D. degrees from the Royal Melbourne Institute of Technology (RMIT), Australia. He was a part of the Technical Services Group, Ericsson Australia, and a Research Engineer at RMIT University. He is currently a Professor of information technology and the Deputy Head of the School of Information Technology, Deakin University, Australia. He leads a team of researchers and Ph.D. students in the broad areas of communication systems and cybersecurity with a focus on emerging domains, such as the IoT, pervasive computing, applied machine learning, and ambient intelligence. His research has been funded by the National Security Science and Technology Branch of the Office of National Security in collaboration with the Defence Signals Directorate, the Australian Research Council, and industry partners. He is the Founding Chair of the Future Network Systems and Security Conference series. He is also an Associate Editor of the journal of Cyber-Physical Systems.