Decentralized position detection for moving vehicles

Francesco Pollicino*, Samih Eisa†, Pedro Rosa†, Miguel L. Pardal†, and Mirco Marchetti*
* Department of Engineering "Enzo Ferrari", University of Modena and Reggio Emilia
{francesco.pollicino, mirco.marchetti}@unimore.it
† INESC-ID, Instituto Superior Técnico, Universidade de Lisboa
samih.eisa@inesc-id.pt, {pmgsrosa, miguel.pardal}@tecnico.ulisboa.pt

***Abstract*—Modern cars are equipped with sensors that can detect other moving vehicles and obstacles on the road. However, their range is usually limited to line-of-sight and their accuracy is also limited. To provide information beyond the sensor range, each vehicle broadcasts Basic Safety Messages (BSMs) with its position and speed. For road awareness, it would be best if multiple vehicles could confirm the position (*redundancy*), using their on-board sensors for verification (*diversity*), and excluding position and speed errors (*plausibility*). This paper presents a decentralized solution that uses multiple vantage points to provide more trust in moving vehicle position data. It extends broadcast messages with sensor verification and plausibility filtering. It processes a *stream* of data from nearby vehicles and for short time periods, to achieve the safety benefits without the privacy risks of long-term data retention. The proposal was evaluated with detailed simulations with different levels of traffic and misbehavior. It provides good detection results with only a limited increase in network and computing resources.**

***Index Terms*—Vehicle positioning, Location proof, VANET, V2V, C-ITS, BSM, DSRC, WAVE**

I. Introduction

Vehicular Ad-hoc Networks (VANETs) [1] are designed to improve the automotive driving experience through communication among roadside infrastructure, road users, and vehicles. VANETs must support heterogeneous environments and must satisfy strict constraints for communication, such as: low latency, security, and dynamic network reconfigurations. Once a VANET is in place, Vehicle-to-Vehicle (V2V) communications allow the broadcasting of periodic Basic Safety Messages (BSMs) to announce vehicle position and speed, with transmission range up to 1000 meters in ideal conditions, beyond line-of-sight. The increasing adoption of VANETs will enable a new set of applications to reduce fuel/energy consumption and increase travel comfort and *safety*. Some examples of the latter are: blind spot warnings, do-not-pass warnings, intersection crossing assistance, and general lane/road problems. Typically, in these applications, each vehicle will receive messages through the VANET, using its sensors and plausibility filters to decide if there are any necessary changes to the status (speed or direction) or if it is necessary to alert the driver about some unexpected event. Another use case that can benefit from position verification is *platooning* [2] where the participants need to verify membership of other vehicles on the same road.

Despite its potential, V2V data sharing also raises security concerns. External security for a VANET is given by using cryptographic keys and algorithms to protect the exchanged messages, supported by a Public Key Infrastructure (PKI). Internal security is handled by various misbehavior detection systems [3] that detect incongruences in a series of position messages received by the vehicles, but they have only a local perspective. Current state-of-the-art anomaly detectors cannot detect some attacks [4], [5] and existing techniques for position verification [6], [7] are not directly applicable in VANETS due to the high mobility and speed of the nodes.

In this paper, we propose a novel mechanism for position verification of moving vehicles. The core idea is to extend the reach of vehicle awareness with *transitive sensor readings*, i.e., relying on sensors from other vehicles to confirm positions. This turns each vehicle into a vantage point for observation that verifies the position of vehicles with its sensors and extends BSM with a list of surrounding vehicles. Once received, the vehicle positions can be tagged as *trusted* (if confirmed by sensors), *plausible*, or as *not trusted*. This classification can be used to improve decision-making in the road safety use cases. We named the system *Miradouros* as it is the Portuguese word for privileged observation spots.

The rest of the paper is organized as follows. Section II describes the base knowledge required for understanding the paper. Section III describes the details of the proposed solution and its applicability is simulated in Section IV. Section V describes other solutions for similar problems. The paper concludes in Section VI.

II. Background

This section presents a description of VANETs along with solutions to increase the security of V2V communications.

Many standards have been proposed for regulating vehicular communications, but the two with more momentum are the Wireless Access in Vehicular Environments (WAVE) for USA and the ETSI ITS-G5 for Europe. Both standards are based on the IEEE-802.11p [8] for the physical and medium-access layers. IEEE-802.11p uses frequencies from 5.850 to 5.925 GHz supporting 3 and 27 Mbps in a 10 MHz channel bandwidth and 6 and 54 Mbps in a 20 MHz channel bandwidth. Both standards support the broadcast of basic V2V safety messages every 100 ms containing the core vehicle data to provide situational data to surrounding vehicles up to 1000 meters (even in urban areas), with a maximum relative vehicle speed of 110 km/h. In particular, for safety communications, WAVE uses the IEEE Basic Safety Message (BSM), while ETSI ITS-G5 uses the ETSI Cooperative Awareness Message (CAM). BSM and CAM messages contain information about

the vehicle, such as its position (latitude and longitude), the level of accuracy of the GPS, the status of the braking system, and other physical attributes (such as the length and the width of the vehicle), to prevent or mitigate dangerous situations. In this work, we consider the WAVE standard as reference. Our focus is on the application level, so our proposal can be easily extended and evaluated with the other standards.

Since VANETs are crucial for future V2X safety applications, different security solutions aimed to protect them from malicious activities have been presented in literature. Our discussion focues on two components: SCMS and MDS. The SCMS (Security Credential Management System) is a security solution for V2X communication and uses a PKI (Public Key Infrastructure) for the generation and distribution of the cryptographic certificates required by the vehicles to ensure the authenticity and integrity of the communications. However, given the scale and scope of the system, it is unrealistic to assume that anyone with a valid cryptographic certificate should be trusted. The MDS (Misbehavior Detection System) is responsible for evaluating the truthfulness of the messages sent by vehicles with valid certificates, and reporting suspicious activities to an MA (Misbehavior Authority), that can later issue certificate revocation requests. To guarantee privacy for the communication between the entities and prevent tracking, multiple pseudonyms are assigned to the vehicles [9], [10]. The cooperation between the SCMS and the MDS in securing VANETs communication is extremely important, since the former component provides security guarantees against external attackers (as described in the IEEE 1609.2 security standard [11]), while the latter is responsible for the detection and mitigation of attacks coming from insiders (i.e. vehicles or infrastructure elements).

III. Proposed solution

This section describes the architecture, data structures, and algorithms of *Miradouros*.

A. Decentralized Architecture

In the proposed architecture, each vehicle runs its own instance of the solution, relying on position messages received through the VANET, obstacle detection made by on-board sensors (e.g., cameras, radars, and lidars), and position plausibility rules, akin to misbehavior detection, considering the current and past positions. Each vehicle periodically broadcasts a BSM, e.g., every 100 ms as recommended in the IEEE 1609 WAVE standard. The BSM includes all information related to the current state of the sending vehicle. We assume that each vehicle adds additional information relative to the nearby vehicles' position, as described in Section III-C. Each element does its own computations and does not rely on the computations made by others. As such, there is no centralized source of *truth* as each vehicle broadcasts its own signals, and the positions are computed by taking into account the history and messages received during the given period.

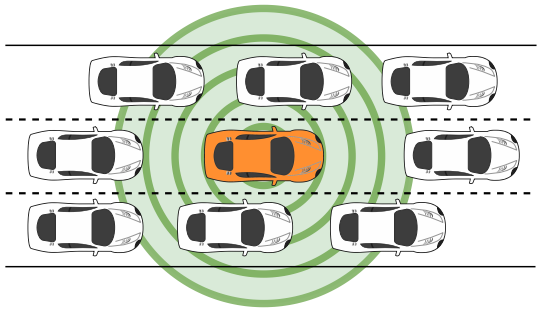

Fig. 1: Sensor range for vehicle moving on a road.

B. Threat model

Miradouros helps detect misreported positions that malfunctioning location systems may cause. However, Miradouros also considers malicious adversaries. The adversary is interested in attacking the position data protected by *Miradouros*. In particular, it aims to convince the other participants of the VANET that it is in a position different from its actual position, probably to hide some traffic violations. If we consider a VANET architecture composed of all the security countermeasures described in Section II, we can classify the adversary as an *insider* attacker following the classification presented in [3]. An insider attack is a typology of attack in which the attacker has acquired valid cryptographic credentials from the PKI to participate in the communications.

Regarding the *privacy* of road drivers, the system is designed to not store persistent information. The messages and perceptions are processed in a short period, relevant for safety applications, but no long-term information is retained.

C. Algorithm

The rationale of the proposed solution is that each vehicle should broadcast additional information about surrounding vehicles (illustrated in Figure 1). In that way, each receiver vehicle can collect more information about the environment and use this information to validate the vehicles' position.

Before describing the algorithm, it is helpful to introduce the two most crucial used data structures: the *Surrounding Vehicle List* (SVL) and *Observed vehicles Table* (ObsTable). Both SVL and ObsTable contain the same category of information (e.g., identifier, position, timestamp). However, the concrete type of information included is a parameter of the proposed algorithm. It can be adjusted to include more information at the expense of the size and network overhead (as discussed in Section IV-A and IV-C). The SVL is populated by each sender vehicle and included in the broadcast BSMs, while each receiver manages the ObsTable. Each vehicle is both a `sender` and a `receiver`, and the two roles are processed simultaneously, so each vehicle processes and manages, at the same time, its own SVL and ObsTable data.

The `sender` role is illustrated in Figure 2. Every cycle, each vehicle produces a standard BSM, appends the SVL information about nearby vehicles, and broadcasts the message.

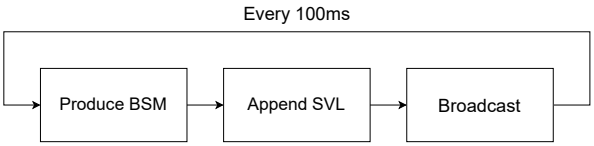

Fig. 2: The flowchart of the `sender` role.

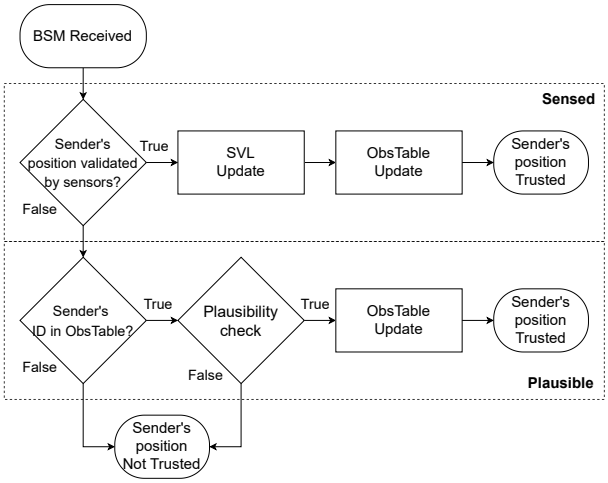

Fig. 3: The flowchart of the `receiver` role.

The SVL is cleared at the end of each cycle (i.e., after each message is broadcast) and populated in the `receiver` role.

The `receiver` role is represented in Figure 3. All the depicted steps are executed for each received BSM. This procedure allows to tag the sender's position included in each BSM as *trusted* or *not trusted*.

When a vehicle receives a BSM, it should first try to validate the sender's position using its own sensors (e.g., camera, radar, lidar). If the vehicle's sensors can confirm the position of the sender, the receiver includes the information of the sender in the SVL (`SVL Update`), updates the ObsTable (`ObsTable Update`), and tags the position of the sender as *trusted*. In this case, the BSM sender's position is considered as *sensed*.

If the sender's position included in the received BSM cannot be verified using the vehicle's sensors, it can be validated in the next step. The receiver should check if the *identity* of the sender is included in the ObsTable, and if yes, use one or more plausibility checks to validate the sender's position. If the position of the sender is validated, the ObsTable is updated, and the position of the sender is tagged as *trusted*. In this case, the BSM sender's position is considered as *plausible*.

Finally, if the sender's position included in the received BSM cannot be validated in either way, the BSM is tagged as *not trusted* but still *received*.

D. Operation

The main difference between a *sensed* and *plausible* position is that only the sensed positions are included in the SVL that will be added to the next BSM, i.e., only sensor confirmations are communicated in the SVL.

Fig. 4: Traffic example: vehicle X overtook C in a curve, crossing a solid line, and is endangering both B and A.

Time	ObsTable State		Messages/Perceptions
T_{n-1}	Sensed:		BSM(B,SVL(A,C,X!))
	Plausible:		BSM(X)
	Received:	B, X, C	BSM(C,SVL(X!,B))
T_n	Sensed:		BSM(B,SVL(A,C,X!))
	Plausible:	B, X, C	BSM(X)
	Received:		BSM(C,SVL(X!,B))
T_{n+1}	Sensed:	B	Sensed(B)
	Plausible:	X!, C	BSM(B,SVL(A,C,X!))
	Received:		BSM(X)
			BSM(C,SVL(X!,B))

TABLE I: Processing steps for vehicle A: *Sensed* is more trusted than *Plausible* that is more trusted than *Received*. X! is the actual position of X.

Let us consider the traffic example represented in Figure 4 and the operation steps of *Miradouros* shown in Table I. In this example, we have the vehicle X doing a dangerous overtaking on a road where the maneuver is prohibited due to a curve and poor visibility. Also, X is an adversary, misreporting its position to hide the traffic violation.

In Table I we can see a representation of the operation of *Miradouros* in vehicle A.

At the time T_{n-1}, the vehicle A became aware of the position of vehicles B, X, and C, thanks only to V2V communications. However, the position of these vehicles is not yet validated. In fact, the position of X is not correct. At this time, the contents of the SVL from both B and C are still being ignored because B and C are out of the sensors' range.

At time T_n, vehicle A receives updated positions. Since the changes are physically consistent with their previous position, the new positions are *plausible*. At this time, the position of B is still not confirmed by A's sensors.

At time T_{n+1}, A can now sense B. Since the position of B is validated, A can now accept the SVL contents of B and

update its own ObsTable with $X!$ and C received within the BSM sent by B. This is critically important, as the position of $X!$ is now updated, and the dangerous situation is detected.

IV. Results

This section presents the assessment of *Miradouros*, based on extensive simulations.

A. Simulation Setup

For the simulation, we used a subsection of the LuST scenario [12], that includes 84 km of roads with different levels of traffic (300, 400, 500, 700, and 1000 vehicles), and we consider 5% level of misbehavior in simulation scenarios. In particular, we consider as misbehavior the set of possible faulty behaviors described in [13] in which a faulty or misbehaving vehicle broadcasts an incorrect position. In our simulation, the misbehaving vehicle broadcasts wrong or random positions from the playground after a random time for a random period. We simulated the considered scenarios by using VEINS [14], an open source framework for vehicular network simulations based on OMNET++ [15] for the simulation of networks and on SUMO [16] for the simulation of the road traffic. In our simulation, we adopted the IEEE 802.11p, IEEE 1609.4 DSRC/WAVE [17] module to extend the VEINS capabilities in a way to enable the DSRC/WAVE stack, Quality-of-Service channel access, Wave Short Message (WSM) management and periodic beaconing of BSMs. We also used the Physical Layer [18], Obstacle Shadowing [19], and Antenna Patterns [20] modules to simulate the propagation and attenuation of the wireless signals to recreate proper signal coverage of messages in urban environments. The vehicles simulated in our scenarios are programmed to send beacon messages every 100 ms, as recommended by the SAE J2945-201712 standard [21]. For the simulation of the vehicle's sensors, we suppose that a vehicle can percept the surrounding environment in a range of 100 meters, in all directions, with an accuracy of 80%, so the receiver can *validate* the sender with its sensors.

As described in Section III-C, the proposed solution can be tuned with different parameters. In this paper, we consider the following configuration. The lifetime of all entries in the ObsTable is set to 20 cycles, so an entry in the table is considered expired if not updated for more than 2 seconds. As plausibility check, we used a consistency verification based only on the position of the vehicles. We consider two positions plausible if, given a maximum speed $maxspeed$ and the $timespan$, the distance between the two points can be traveled by a vehicle with speed $maxspeed$. We set the maximum vehicle's speed in the simulation to be 55 meters/second (approximately 200 Km/hour). Again, we remark that the accuracy of the sensors, the expiration time of the ObsTable, and the plausibility check algorithm are only parameters of the proposal and can be tuned for more specific applications.

B. Simulation Results

The simulations were run and produced 60 seconds of data, summarized in Table II. The rows report different `Runs`, and the columns report the average results for each vehicle. For each `Run` we indicate the number of vehicles (`#V`), the number of malicious vehicles (`#MV`), the average percentage of sensed positions (`Sensed`), the average percentage of ObsTable hit of each vehicle (`ObsTable Hit`), and the average percentage of ObsTable miss of each vehicle (`ObsTable Miss`). We have an ObsTable hit if the sender of a received BSM is already in the table and a miss if it is not.

The results show that we have more than 50% of probability (`Sensed` + `ObsTable Hit`) that a received BSM can be validated and that probability increases in scenarios with more vehicles. The probability that the sensors validate a BSM is stable in all the different densities (`#V`).

The columns True Negative (`TN`) represent the percentage of BSMs correctly tagged as *not trusted*. We consider as `TN` both a malicious BSM with the sender ID that is not included in the table (Malicious ObsTable Miss `MTM`) and a BSM with a sender id included in the table but tagged as non-plausible. The column False Negative (`FN`) represents the percentage of BSMs that are tagged as *trusted* but that are malicious, and ObsTable Hit False Positive (`THFP`) the percentage of BSMs with table hit tagged as *not trusted* but that are genuine. Finally the columns `Sent SVL Size` and `Recv SVL size` report the average size of the SVL for sender and receiver.

The results show that we have about 100% detection rate of the malicious BSMs in all scenarios, while the rate of `THFP` increases in more dense scenarios. A message is classified as THFP by the plausibility check, so improved plausibility algorithms will help to reduce the false positives.

We tag the ObsTable Miss messages as *not trusted* because we do not have enough evidence to classify them as *trusted*.

We do not consider this a disadvantage because we prefer to have high reliability instead of a high recall, i.e., we prefer not to tag a malicious message as trusted at the cost of losing some genuine messages. Since the cycles are short and the BSMs are repeated, the positions will eventually be detected.

In Table III we further investigate if the messages included in the ObsTable Miss column of Table II can be useful in some scenarios. In particular we consider the communication requirements defined by the NHTSA [22], which considers multiple vehicle communication scenarios and defines the constraints that must be satisfied to guarantee safety.

The report identifies 8 high-priority and safety-critical scenarios, and for each of them defines the allowable *latency* and communication *range*, respectively:

- Pre-Crash Sensing: 20 ms, 50 m;
- Traffic Signal Violation Warning: 100 ms, 250 m;
- Curve Speed Warning: 1000 ms, 200 m;
- Emergency Electronic Brake Light: 100 ms, 300 m;
- Cooperative Forward Collision Warning: 100 ms, 150 m;
- Left Turn Assistant: 100 ms, 300 m;
- Lane changing Warning: 100 ms, 150 m;
- Stop Sign Movement Assistance: 100 ms, 300 m.

Following the setup proposed in this paper (c.f. Section IV-A), we consider the distances less than 100 m managed by the vehicle's sensors. Table III reports for each `Run` the

Run	#V	#MV	Sensed	ObsTable Hit	ObsTable Miss	TN MTM	TN Plausibility	FN	THFP	Sent SVL Size	Recv SVL Size
1	300	5% (15)	35%	21%	44%	97%	3%	0%	20%	2.23	3.17
2	400	5% (20)	37%	23%	40%	95%	3%	2%	23%	2.45	3.63
3	500	5% (25)	36%	26%	38%	96%	4%	0%	25%	3.68	4.70
4	700	5% (35)	36%	31%	33%	95%	5%	0%	29%	5.20	6.80
5	1000	5% (50)	36%	33%	31%	96%	3%	1%	30%	7.13	8

TABLE II: Results for simulations with a duration of 60 seconds (600 BSM cycles).

percentage of ObsTable Miss with different distances between the sender and the receiver.

Run	#V	ObsTable Miss			
		100-150 m	150-200 m	200-250 m	250-300 m
1	300	13%	20%	19%	11%
2	400	13%	20%	15%	10%
3	500	9%	17%	17%	13%
4	700	8%	15%	22%	10%
5	1000	7%	14%	19%	14%

TABLE III: Analysis of the distances of ObsTable Miss.

The percentage of table miss in the short-medium distances (100-150 m and 150-200 m) decreases with an increase of the number of vehicles, showing that our proposal works better in a scenario with more dense traffic. In medium-long distances (200-250 m and 250-300 m), there is no strong correlation between the percentage of table misses and the number of vehicles. In fact, the value of table miss for the 200-250 m distance range is included between 15% and 22%, while for the 250-300 m distance, between 10% and 14%. For all the scenarios we have, there is about a 0% of table miss for distances < 100 m because these distances are validated mainly by the vehicles' sensors, while we have more than a 45% of table miss for messages with positions over 300m in all the scenarios. Since most of the table misses are for medium and long distances (> 200 m), an application that relies on the trustness of the positions can delay the decision of non-safety critical maneuvers to the following cycles.

C. Applicability of the proposed solution

Based on the results reported in [23] and specifications of the NHTSA report [22], the average size of a signed BSM that follows the security recommendation of the IEEE 1609.2 standard [11] is 460 bytes, and with a network bandwidth of 6 MiB/s the maximum number of messages supported by the network is 163 messages. The proposed solution requires each vehicle to add additional information about the positions of the surrounding vehicles, causing an increment in the size of the BSMs and an extra overhead on the entire network. In particular, we consider that each row of the SVL and the ObsTable contains an ID, a $position$, and a $timestamp$. Let us consider that the reported position is expressed using standard GPS coordinates using one word of 32 bit for each field. We can assume that the overhead is 64 bit (or 8 bytes) for each coordinate included in the BSM, 4 bytes for the identifier, and other 4 bytes for the timestamp.

As described in Section III-C, each vehicle should include the position of a surrounding vehicle in a BSM only if this position was validated by its sensors. In a worst-case scenario, like the one depicted in Figure 1, we can suppose that the maximum number of positions that a vehicle can include is 8, which gives an overhead of (8+4+4)*8 = 128 bytes, so a BSM of size 590 bytes and a reduction of network capacity from 163 to 127 messages in term of the size of the SVL, that is enough to support the scenarios proposed in [23]. However, from the results in Table II, we can see that, on average, the size of the SVL depends on the traffic density, with a value from 2 and 3 for the less dense scenario to 7 and 8 for the more dense scenario for the sent and received SVL, respectively. These results show sustainable network overhead in all scenarios.

The size of the ObsTable that each vehicle should use depends on the number of received BSMs and the expiry time of each row. The time required for a lookup is $O(1)$, on average, assuming the use of hashtables.

V. Related Work

Sharma et al. [24] present a novel machine-learning approach for detecting false positions in BSMs. Several BSMs are used to combine information and get more reliability. The authors claim to detect several falsification attacks with the increased knowledge of groups of BSMs using Machine Learning (ML). The ML approach has the potential to detect a broader set of attacks than Miradouros. However, this approach would delay the time taken for detecting the same subset of attacks that Miradouros does due to the faster local management of BSMs with in-vehicle processing.

Ilango et al. [25] propose NPFADS (Novel Position Falsification Attack Detection System) that can learn and detect novel position falsification attacks using several RSU and fog computing nodes. These nodes are connected to the Internet to detect emerging attacks on the network using BSM. Unlike Miradouros, the authors use a distributed set of components to share information about attacks inside the network. This system provides a larger knowledge of attacks in several locations, being able to share that information between distant vehicles However, this approach requires increased infrastructure support and can also increase processing needs in vehicles, unlike our solution that works on a local context only.

Dokur et al. [26] demonstrated three V2V safety applications (Front Collision Warning, Emergency Electronic Brake Light and Blind Spot Warning) using only exchanged BSMs to accurately predict the relative positions of vehicles. They do not rely on camera-based sensors due to the less reliability of such devices under special weather conditions. However, this solution can leave out the use of valuable information taken

by car sensors and delay the time taken for safety applications like emergency braking. Miradouros runs an instance of the solution in each vehicle, using todays state-of-the-art on-board sensors and can provide a faster perception of traffic to trigger safety actions.

Tsai et al. [27] propose a solution for enhancing vehicles position using relative positioning instead of relying only on the GPS position. Like Dokur's work, this work aims to provide drivers with a braking warning signal without the obscurity caused by weather using V2V communication. However, this work has privacy implications that do not affect our proposal, because it uses and stores license plate numbers to identify vehicles in BSMs. Miradouros stores sensible information, i.e, vehicle identities, but only for a short period, soon discarding all used and unnecessary information about vehicles.

VI. Conclusion

This paper presents a novel decentralized proposal for the validation of the position of moving vehicles in VANETs. As a first contribution, we propose a distributed algorithm that fuses the perception of the enviroment that each vehicle is able to construct thanks to its sensors, with the information collected from all the received BSMs broadcast by each nearby vehicle. As a second contribution, we performed an evaluation of the proposal using simulations of realistic scenarios, using different levels of traffic and misbehavior in position reporting. These results have shown that we can obtain good detection results making our proposal suitable for road safety applications that rely on the correct position of vehicles.

Acknowledgements

This work was supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UIDB/50021/2020 (INESC-ID).

References

[1] C. K. Toh, *Ad Hoc Wireless Networks: Protocols and Systems*, 1st ed. USA: Prentice Hall PTR, 2001.

[2] M. Asplund, "Model-based membership verification in vehicular platoons," in *2015 IEEE International Conference on Dependable Systems and Networks Workshops*, 2015, pp. 125–132.

[3] J. Kamel, M. Ansari, J. Petit, A. Kaiser, I. Ben Jemaa, and P. Urien, "Simulation framework for misbehavior detection in vehicular networks," *IEEE Transactions on Vehicular Technology*, 2020.

[4] F. Pollicino, D. Stabili, G. Bella, and M. Marchetti, "Sixpack: Abusing abs to avoid misbehavior detection in vanets," in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*. IEEE, 2021, pp. 1–6.

[5] J. Lastinec and M. Keszeli, "Analysis of realistic attack scenarios in vehicle ad-hoc networks," in *7th International Symposium on Digital Forensics and Security (ISDFS)*. IEEE, 2019.

[6] S. Gambs, M. Traoré, M. Roy, and M.-O. Killijian, "Props: A privacy-preserving location proof system," *Proceedings of the IEEE Symposium on Reliable Distributed Systems*, vol. 2014, 10 2014.

[7] J. Ferreira and M. L. Pardal, "Witness-based location proofs for mobile devices," in *2018 IEEE 17th International Symposium on Network Computing and Applications (NCA)*. IEEE, 2018, pp. 1–4.

[8] IEEE 802.11 Working Group and others, "Ieee standard for information technology–telecommunications and information exchange between systems–local and metropolitan area networks–specific requirements–part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 6: Wireless access in vehicular environments," *IEEE Std*, vol. 802, no. 11, 2010.

[9] L. Buttyán, T. Holczer, and I. Vajda, "On the effectiveness of changing pseudonyms to provide location privacy in vanets," in *European Workshop on Security in Ad-hoc and Sensor Networks*. Springer, 2007.

[10] G. Calandriello, P. Papadimitratos, J.-P. Hubaux, and A. Lioy, "Efficient and robust pseudonymous authentication in vanet," in *Proceedings of the fourth ACM international workshop on Vehicular ad hoc networks*, 2007.

[11] IEEE, "Standard for Wireless Access in Vehicular Environments–Security Services for Applications and Management Messages," IEEE, Std. 1609.2a-2017, 2017.

[12] L. Codeca, R. Frank, and T. Engel, "Luxembourg sumo traffic (lust) scenario: 24 hours of mobility for vehicular networking research," in *2015 IEEE Vehicular Networking Conference (VNC)*. IEEE, 2015.

[13] J. Petit and R. Ansari, "V2x validation tool," *BlackHat 2018*, 2018.

[14] C. Sommer, R. German, and F. Dressler, "Bidirectionally coupled network and road traffic simulation for improved ivc analysis," *IEEE Transactions on Mobile Computing*, 2011.

[15] A. Varga and R. Hornig, "An overview of the omnet++ simulation environment," in *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

[16] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. WieBner, "Microscopic traffic simulation using sumo," in *The 21st IEEE International Conference on Intelligent Transportation Systems*, 2018.

[17] D. Eckhoff, C. Sommer, and F. Dressler, "On the necessity of accurate IEEE 802.11 p models for IVC protocol simulation," in *2012 IEEE 75th Vehicular Technology Conference (VTC Spring)*. IEEE, 2012, pp. 1–5.

[18] F. Bronner and C. Sommer, "Efficient multi-channel simulation of wireless communications," in *2018 IEEE Vehicular Networking Conference (VNC)*, 2018.

[19] C. Sommer, D. Eckhoff, R. German, and F. Dressler, "A computationally inexpensive empirical model of ieee 802.11 p radio shadowing in urban environments," in *Eighth international conference on wireless on-demand network systems and services*, 2011.

[20] D. Eckhoff, A. Brummer, and C. Sommer, "On the impact of antenna patterns on vanet simulation," in *2016 IEEE Vehicular Networking Conference (VNC)*, 2016.

[21] SAE International, "Dedicated short range communications (dsrc) message set dictionary," *SAE International*, 2016.

[22] National Highway Traffic Safety Administration, "Vehicle safety communication projet – final report," NHTSA, DOT HS 810 591, Apr. 2006.

[23] F. Pollicino, D. Stabili, L. Ferretti, and M. Marchetti, "Hardware limitations to secure c-its: Experimental evaluation and solutions," *IEEE Transactions on Vehicular Technology*, 2021.

[24] A. Sharma and A. Jaekel, "Machine learning based misbehaviour detection in vanet using consecutive bsm approach," *IEEE Open Journal of Vehicular Technology*, vol. 3, pp. 1–14, 2022.

[25] H. S. Ilango, M. Ma, and R. Su, "A misbehavior detection system to detect novel position falsification attacks in the internet of vehicles," *Engineering Applications of Artificial Intelligence*, vol. 116, p. 105380, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952197622003876

[26] O. Dokur and S. Katkoori, "Three connected v2v applications based on dsrc basic safety messages," in *2022 International Conference on Connected Vehicle and Expo (ICCVE)*, 2022, pp. 1–6.

[27] M.-F. Tsai, Y.-C. Chao, L.-W. Chen, N. Chilamkurti, and S. Rho, "Cooperative emergency braking warning system in vehicular networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, 12 2015.