Spoofed Data Detection in VANETs using Dynamic Thresholds

Jonathan Petit, Michael Feiri, Frank Kargl
Distributed and Embedded Security Group
University of Twente, The Netherlands
Email: j.petit — m.feiri — f.kargl@utwente.nl

***Abstract*—Vehicular ad hoc networks aim at enhancing road safety by providing vehicle-to-vehicle communications and safety-related applications. But safety-related applications, like Local Danger Warning, need a high trust level in received messages. Indeed, decisions are made depending on these messages. To increase the trustworthiness, a consensus mechanism is used. Thus, vehicles make a decision when a *threshold* is reached. Setting this threshold is of main importance because it impacts the decision delay, and thus, the remaining time for a driver reaction. In this paper, we investigate the problem of threshold establishment without globally unique identifier system (GUID). We propose to model the threshold as a Kalman filter and provide an algorithm to dynamically update the threshold. By simulations, we investigate the problem of insider attackers that generate information forgery attacks. Simulation results show that our dynamic method suffers from a bootstrapping phase but reduces the percentage of wrong decisions. Nevertheless, as future work, further analysis of default threshold value will be done.**

***Index Terms*—consensus, spoofing detection, dynamic threshold, VANET.**

I. Introduction

One of the primary motivations for research on inter-vehicular communication is deployment of safety applications such as cooperative collision avoidance, local danger warning, and road hazard notification. By wirelessly exchanging information on mutual positions, speed, and heading, the basic idea is to provide a better situational awareness for close-by vehicles so that local applications can decide whether there are potentially critical situations with a risk of collision or crash. These applications would then provide warnings to drivers in such critical driving situations. Expectations are that this will significantly reduce the numbers of traffic-related accidents and injuries.

However, this promise can only be fulfilled if the system works with very high reliability and this, consequently, requires resilience against security attacks. It is crucial to ensure that situation information exchanged between vehicles cannot be forged or modified by an attacker. If you would assume that an attacker can provide wrong position or speed information to other vehicles, this will very easily lead to wrong or suppressed warnings and thus to inappropriate behavior of drivers. For example, a driver that is warned about an immediate crash ahead will likely break sharply and might cause rear-end accidents as an effect.

Without proper security mechanisms in place, inter-vehicular networks are especially vulnerable to such false data injection attacks where misbehaving vehicles inject erroneous information into the network for selfish or malicious reasons. Basic security mechanisms for Vehicular Ad hoc Networks (VANETs) suggest to use authentication and integrity protection mechanisms to ensure that only valid vehicles or road-side units participate in communication. This can be implemented using digital signatures and a Public-Key-Infrastructure (PKI) [1] as foreseen by all current standardization efforts [2]. But even in this case, one can still not trust that all (valid) vehicles report correct information [3].

Vehicles will typically receive information from multiple neighbors in their immediate surrounding. In this paper, we assume that this happens by means of Wave Safety Messages (WSMs) as defined in [2]. Assuming that a certain fraction of vehicles is malicious and will report wrong data, this leads to the classical Byzantine Agreement (BA) problem [4] where some vehicles report a problem and some do not, but you do not know which are honest. A closely related sub-problem, the consensus problem, has been extensively studied in general distributed systems [5]. However, in contrast to general distributed systems, we are dealing with a very dynamic environment that requires near real-time decision making while at the same time facing bandwidth constrained communication channels.

In this paper, we are addressing the problem how to determine whether information about an event like icy road or an accident ahead is trustworthy or not. We assume that we receive information from multiple communication partners, some of which might be malicious. Applying a consensus mechanism allows the local On-Board Unit (OBU) to reach a decision before taking actions like warning the driver. Generally speaking, the OBU would require to receive a certain number of consistent reports about a specific event before a warning would be issued. Having such a consensus mechanism in place increases the trustworthiness of received warnings at the expense of additional delay as the OBU would first have to wait for reception of a certain number of messages to reach a certain *confidence threshold* [6].

A question that has been neglected by research so far is how to set this *threshold*. The chosen value will have an influence on a number of parameters like the required processing resources for checking the messages [7]. But most

978-1-4673-0047-6/11/$26.00 ©2011 IEEE

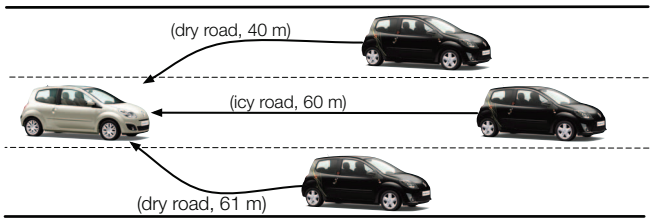

Fig. 1. Problem of deciding on conflicting event notifications

important, it influences the trade-off between the decision delay (and thus the delay until a driver gets warned) and the trustworthiness of the information (and thus the opportunity for an attacker to cheat). So a threshold must be chosen carefully.

There are a number of sub-problems that need to be addressed. First, an OBU needs to decide whether two event warnings received from neighboring vehicles relate to the same event and are subject to consensus checking or not. Assume, for example, that the OBU receives warnings from three vehicles A, B and C, where A reports free road 40 meters ahead. However, B reports icy road 60 meters ahead and C reports dry road conditions 61 meters ahead (cf. Fig. 1). Should those three reports be treated as one event — having a 2:1 majority for dry road ahead — or should we consider A separate and just look at the reports from B and C to check whether there is icy road or not. In the latter case, there is a 50% chance that the car will find icy road after 60 meters.

Current work on trust provisioning and consensus often assumes a unique event identifier and a perfect synchronization between vehicles to work around this problem. We think that such an assumption is not realistic. We will discuss how a consensus mechanism for VANETs can work without such identifiers.

A second issue is that a static threshold will not be sufficient. Depending on driving situations, type of event, previously received information, general context information, or possible reaction to warnings, a different level of trustworthiness might be required before reaching a consensus decision.

By analogy, when a driver drives down an unknown road, he will likely react immediately to any warnings he receives [8]. For example, when a driver A sees an upcoming vehicle flashing its headlights, A will assume that there is a problem or danger ahead and will likely react by slowing down. However, if A does not detect a hazard after a certain distance, he will conclude that it was a false warning or that the problem has disappeared. So, if later (at least after this certain distance), there is another vehicle flashing its headlights, then A might be less responsive and might only respond if two or more vehicles warn him. We use the same idea in our approach. Vehicles will constantly adjust their decision threshold by constantly computing the average "noise level" representing the probability of a false warning (implying the average level of attackers that might send wrong information).

By not assuming a unique event identifier and by having an adaptive threshold scheme, our approach has significant advantages over earlier proposals. It is more flexible and practical while still providing good detection capabilities for spoofed information. Before going into details of our approach, we first discuss related work in section II. In section III, we present the assumptions, a system, and an attacker model. Then, section IV presents a general model for our decision method and the algorithm used to adaptively determine the threshold. Simulation parameters are presented in section V and results are analyzed in section VI. Finally, we conclude our paper with a summary and outlook on possible enhancements and open problems.

II. Related Work

Setting consensus parameters is a trade-off between detection power and overhead. In [9], Ostermaier *et al.* proposed a decision method named *Majority of freshest X with Threshold*. In this decision method, if the number of same warnings is higher than the *Threshold*, then the vehicle decides according to the majority of the *X* last warnings. Their simulations concluded that the "*majority of freshest X with Threshold*" is the method best suited to provide protection against information forgery attacks in VANETs. But they did not consider how to set the *Threshold* or *X*, and leave this issue open.

Hyunjin *et al.* proposed a model to distinguish spurious messages from legitimate messages [10]. They used a threshold and a value of *certainty of the event* to decide when to warn the driver. To compute the *certainty of the event*, six sources of information are used (digital signature, source location, local sensors, WSM, infrastructure, sender reputation). They applied this basic framework to the Emergency Electronic Brake Lights application.

Leinmüller *et al.* [11] proposed a cooperative position verification to detect position cheating. Authors detailed autonomous and cooperative sensors that use map-based verification, acceptance range threshold or exchange of neighbor tables to estimate the trustworthiness of other nodes' position claims. However, the paper does not address the question how thresholds could be adapted to achieve higher accuracy.

Aforementioned studies propose different decision methods but assume a globally unique identifier system. Dietzel *et al.* proposed a fuzzy logic strategy to link multiple warnings to the same event [12]. This mechanism alleviates the need of a Global Unique Identifier system (GUID). Indeed, even if vehicles are not fully synchronized and report event with slightly different coordinates, the receiver could determine if warnings correspond to the same event or not. Thus, vehicles do not need an unique warning identifier to identify the event.

In this paper, we propose a methodology to set the consensus parameters assuming no globally unique identifier system. More specifically, we investigate the problem of spoofed data detection in VANETs using a dynamic majority voting scheme.

III. System model

A. Assumptions

We assume vehicles drive on a multi-lanes highway which use a typical safety-related application for V2V communi-

cation: the Local Danger Warning (LDW). In LDW [13], vehicles exchange information about dangerous traffic situations based on local sensor readings to realize a collaborative and predictive situation-awareness. As mentioned in [9], a cooperative local danger warning application comprises three steps: detection process, message dissemination, and decision process. In the detection process, vehicles detect hazards with their on-board sensors while driving. Whenever a critical condition is detected, the vehicle triggers the dissemination process and broadcasts a warning message—sent in a WSM every 100 ms [2]. Vehicles receiving such a message, trigger the decision process. If there is sufficient evidence for a critical road condition on the route ahead, the system notifies the driver to have him take appropriate actions.

We are interested in the decision process, where the LDW application has to decide whether or not to take action or notify the driver, because leading the system into a wrong decision is one of the major threats. For example, a relevant decision for a vehicle approaching an accident location on a highway would be to change lane. But, if the system provides an irrelevant warning, the situation will get worse (over-accident and passenger injuries). Moreover, if the system leads to a wrong decision, the driver will not trust the system anymore.

In this paper, we denote as *event* the detection of a hazard (fake or not), and as *source* the first originator of the warning (fake or not). We assume a one-hop broadcast communication and a penetration rate of 100% (i.e. all vehicles are equipped with a DSRC device). Concerning the possible set of decisions for a vehicle, we divide it in two subsets:

- *Vehicle action*: brake, change lane, change path, accelerate, warning light, do nothing. This type of action could be a response to a driver alert.
- *Network action*: broadcast a message, do nothing.

Thanks to the geographical coordinates of the event contained in the WSM, a vehicle could detect a false warning when it passes the warning location. Thanks to the *beaconing* [14] [15], the vehicle has a local view of its neighborhood. Hence, we assume that the vehicle has a spatial representation and could define what is *ahead of* and *behind* it (thanks to geospatial coordinates and a road map). In contrast to previous studies, we do not assume a global unique identifier to be associated with warnings and we also do not assume perfect time synchronization of vehicles. As a result, vehicles will not be able to differentiate whether warnings in different messages relate to the same or two different events. To address this issue we use the fuzzy logic strategy proposed in [12]. We complete the decision method proposed in [16] by adding an aggregation module between the *classifier* and the *dispatcher*. This module is in charge of aggregating multiple warnings as described in [12]. In this paper we focus on modeling the *decision maker* as a Kalman filter to dynamically set the *confidence threshold*.

B. Attacker model

We consider the *information forgery attack*, i.e. an attack where intruders generate false warnings to trigger false decisions trying to manipulate road traffic. We assume that attackers can collaborate to make the fake warning faster accepted (which assumes that attackers know each other). When an attacker receives a fake warning from another attacker, it corroborates the event by generating a fake warning.

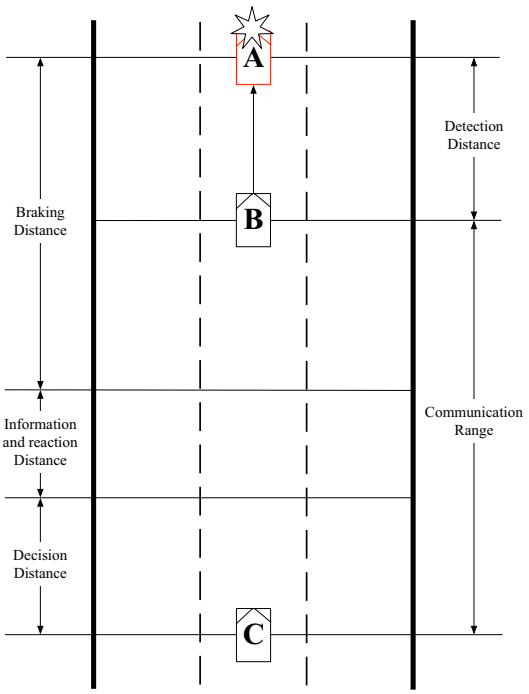

Fig. 2. Definition of distances

C. Definition of the OBU distances

Fig. 2 shows the different distances considered in this system model.

- *Detection distance*: distance where a vehicle could detect the warning with its on-board sensors. It depends on the maximum sensor range.
- *Braking distance*: the braking distance is computed from the current speed of the vehicle, the road condition (dry, wet, snowy), and the vehicle characteristics (tires pressure, brake capacity).
- *Information and reaction distance*: distance for warning the driver which depends on the driver reaction time.
- *Decision distance*: distance allowed for collecting WSMs and making a decision.

In Fig. 2, A is endangered and B detects A thanks to its on-board sensors. Then B broadcasts a WSM. C is in the communication range of B and receives the WSM. On each reception, C computes the braking distance, the information and reaction distance and the decision distance. From Fig. 2, we define the following notation:

- $T_{collision}$: the expected collision time computed by the speed and the distance.
- $T_{braking}$: the time of braking computed by $T_{braking} = \frac{v_k}{a}$, where a is the deceleration rate and v_k the speed of the vehicle V_k.
- $T_{reaction}$: the reaction time of the driver (0.7-1.5 second).

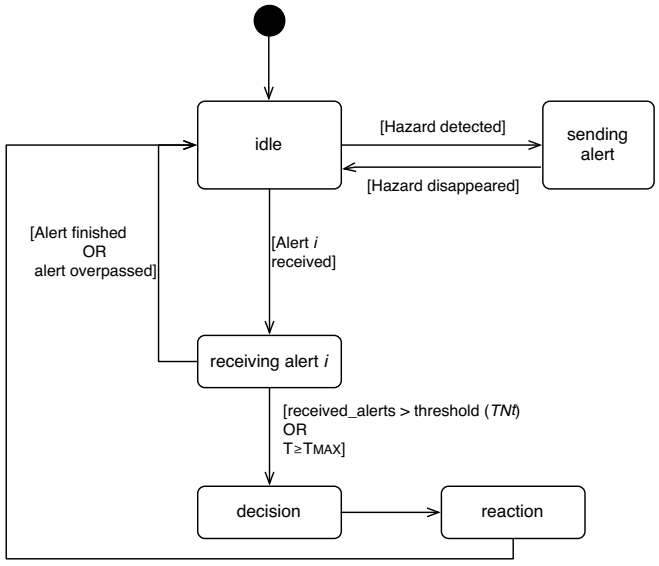

Fig. 3. State transition diagram of the OBU for one alert *i*

- T_{safety}: the time to travels the safety distance, which is computed by $T_{safety} = T_{braking} + T_{reaction}$.
- ΔT_i: the remaining time before the collision, which is computed by $\Delta T_i = T_{collision} - t_i$ (where t_i is the current time).

From Fig. 2, we can compute the maximum time allowed to make a decision before entering the braking distance. One of the goals of this paper is to provide a method to assess the consensus parameters to respect the maximum decision delay allowed. In the following, we define the *decision delay* as the time between the generation of the first warning by the source and the decision by the receiver.

D. State transition diagram of an OBU

Fig. 3 shows the state transition diagram of a vehicle that receives one alert *i*. A vehicle goes from *idle* to *sending alert* when it detects a hazard. The target of the hazard could be itself (the vehicle stops because of an emergency reason), another vehicle or the environment (ice, hole, obstacle). While the hazard is still detected, the vehicle generates an alert. A vehicle goes from *idle* to *receiving alert* when it receives an alert. It keeps collecting WSMs until one of the following conditions is reached. It goes from *receiving alert* to *idle* when the hazard location is overpassed, or when the hazard has disappeared. The vehicle goes from *receiving alert* to *decision* when it receives enough WSMs (i.e. the threshold TN^t is reached). Another case of transition is when the maximum delay allowed before making a decision is exceeded. Indeed, safety-related applications have real-time constraints, and mandate to react before a specified delay (T_{MAX}). T_{MAX} is the maximum time allowed by the application before having critical impact (e.g. an accident). T_{MAX} is computed with the speed, the distance from the hazard and the application design. T_{MAX} is less than 500 ms for highly time-critical applications, and equal to three seconds for time-relevant applications [17]. In Fig. 3, T is the time between the first reception and the current time. Finally the vehicle goes from *decision* to *reaction* state to apply the decision.

IV. Methodology of parameters setting

In this section we want to dynamically estimate the right threshold for making a decision as illustrated in Fig. 3. Parameters involved in the system are details in the next section. Then we detail our methodology to determine consensus parameters.

A. Parameters

We list the parameters involved in our model:

- R: Transmission range.
- $N_{TX}(t)$: Number of one-hop neighbors at time t computed according to [18].
- $Ahead(N_{TX}(t), R)$: *Neighborhood density* that represents the number of neighbors, which are moving ahead of the current vehicle at time *t* in the transmission range *R*.
- $\overline{V^t}$: Neighborhood vector at time t.
- $|\overline{V^t}|$: Cardinality of the neighborhood vector at time t.
- $\overline{A^t}$: Message received vector at time t.
- $|\overline{A^t}|$: Cardinality of the message received vector at time t.
- ξ: *Observation function* to compute the ratio according to the neighborhood vector and the message received vector.
- k: The size of the queue considered for the decision method.
- TN^t: Threshold that corresponds to the average noise level at time t.
- p: The precaution parameter to set the decision point.
- $C_{\lambda(i)}$: Criticalness of the event *i* of type $\lambda(i)$ at time t which is computed from the estimated distance from the hazard, and the vehicle speed.

B. Decision method abstraction: Kalman filter

The Kalman filter is a set of mathematical equations that provides an efficient computational (recursive) means to estimate the state of a process, in a way that minimizes the mean of the squared error [19]. The filter is very powerful in several aspects: it supports estimations of past, present, and even future states, and it can do so even when the precise nature of the modeled system is unknown. When estimation is

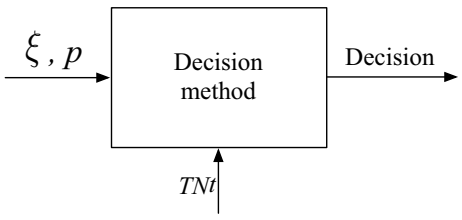

Fig. 4. Kalman filter

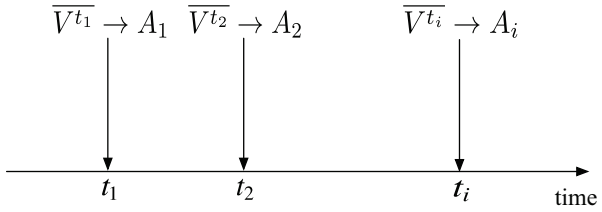

Fig. 5. Discretization of the events

of concern, the Kalman filter is often used. Indeed, it processes all available measurements, regardless of their precision, to estimate the current value of the variables of interest, with use of (1) knowledge of the system and measurement device dynamics, (2) the statistical description of the system noises, measurement errors, and uncertainty in the dynamics models, and (3) any available information about initial conditions of the variables of interest [20].

In this paper, we use the Kalman filter in a very elementary level, where the OBU estimates the state (legitimate or spurious message) based on the received observations. Fig. 4 shows the parameters used in our filter.

- Input: ξ and p.
- Noise: TN^t.
- Decision method (filtering function).
- Output (decision): Result from the decision method.

We detail each of these parameters.

1) Observation function ξ: We discretize the message reception (cf. Fig. 5). A vehicle receives a message A_i from a vehicle member of the neighborhood vector $\overline{V^{t_i}}$ at time t_i. It represents the ratio of messages received to the overall numbers of vehicles present in the vehicle's neighborhood during the interval where those messages were received.

The observation function ξ computes the proportion of messages received compared to the number of neighbors. On each reception, the vehicle computes the ξ function. The vector of value ξ is denoted $\overline{\xi}$ and forms the observation vector used by the filter function. First, we define the observation function by:

$$\xi^1 = \frac{1}{|\overline{V^{t_i}}|} \tag{1}$$

The decision is made from one message and is a function of the neighborhood density.

But, when the vehicle approaches the detection area, the number of messages received increases because the neighborhood density increases. We aggregate the messages received. On receiving a second packet, the vehicle computes:

$$\xi^2 = \frac{2}{|\overline{V^{t_i} \bigcup V^{t_j}}|} \text{ with } t_i < t_j \tag{2}$$

By interpolation, we obtain:

$$\xi^k = \frac{k}{|\bigcup_{j=1}^{k} \overline{V^{t_j}}|} \tag{3}$$

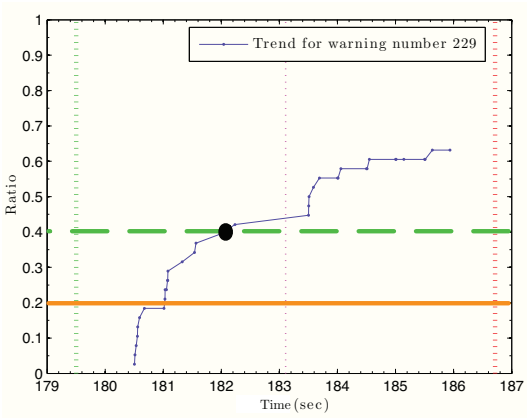

Fig. 6. Example of an observation sequence

The ξ function returns an integer in the interval [0,1] which is denoted as *ratio*. The decision is made according to the entire set of the messages received.

2) Precaution parameter p: When the threshold is reached, the vehicle reacts in function of the precaution parameter (e.g. the dashed horizontal line on Fig. 6). The precaution parameter sets the *decision point* which is the point of the curve when the vehicle reacts (e.g. bold point on Fig. 6). It is based on the criticalness of the event. The criticalness depends on the approximate distance between the vehicle and the danger location [16]. In order to compute the location of the event, we use the following formula:

$$\widehat{(X,Y,Z)} = min(\overline{(X,Y,Z)}) \tag{4}$$

where (X,Y,Z) are the coordinates of a report, $\overline{(X,Y,Z)}$ is the set of reports coordinates and $\widehat{(X,Y,Z)}$ are the closest coordinates from the current vehicle location. Here, we assume that the vehicle uses the coordinates of the closest warning received in the timeframe.

3) Threshold, noise: The threshold TN^t represents the average noise level at time t or the average number of previous false warnings (e.g. the solid horizontal line on Fig. 6). We define it by:

$$TN^t = \frac{FD_t}{|\overline{V^t}|} \tag{5}$$

where FD^t is the number of false detections at time *t*. The false detection is observed where the vehicle passes the estimated location of the event and detects nothing. We detail

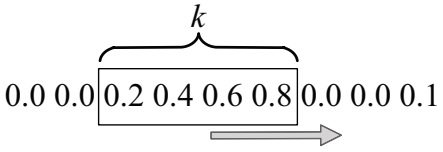

Fig. 7. Concept of sliding window

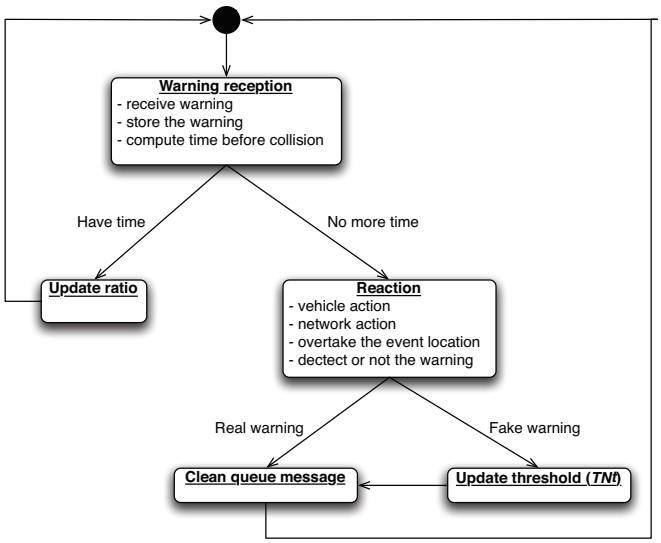

Fig. 8. State machine of threshold determination

the algorithm to update TN^t in section IV-C.

4) Decision method, filtering function: In input, a growing observation vector represents an approach of a potential hazard. The filtering function has to recognize from a k-length observation vector, a sub-sequence of growing values. As shown in Fig. 7, a vehicle receives a series of observation values ($\xi \in [0,1]$) and should detect a pattern of k values that represents an approaching event. k is the sliding window of the considered values for the detection. The main problem is to define the best k, which corresponds to find the X of Ostermaier's method [9].

5) Decision: The decision is chosen among the different reaction options described in section III-A.

C. Threshold determination

The threshold determination is defined as a cyclic state machine (cf. Fig. 8). When a vehicle receives a warning, it computes ΔT_i. If $\Delta T_i > T_{safety}$, then the remaining time permits to collect more messages. The ratio is updated. Otherwise, there is no more time to collect and the vehicle has to make a decision with the current set of messages received. When the vehicle overpasses the danger location and does not detect a danger, it updates the threshold TN^t. Hence the vehicle becomes more suspicious. Whatever the result of the detection, the vehicle cleans its queue message by deleting the warnings that have a coordinate behind it.

V. Simulation Parameters

We use the network simulator ns2.34 [21] to analyze the dynamic threshold strategy. The scenario is a highway with three lanes in one direction, where a percentage of vehicles (20% here) will randomly stop to generate an event. Either for beaconing and event-triggered messages, vehicles send messages of 254 bytes to illustrate a signed message with P-224 [7]. Simulations are run 100 times and results are provided with a 95% interval confidence. We investigate the impact of the percentage of attackers and the neighborhood density on the threshold and the decision delay. Table I details the simulation parameters.

Parameter	Value
Communication range R (m)	300
Density of vehicle (veh/km/lane)	5, 10
WSM frequency (Hz)	10
Simulation time (sec)	300
Propagation model	Nakagami (m=3)
Packet size (bytes)	254
Vehicle speed (m/s)	27.7, 30.5, 36.1
Percentage of attacker	[0,50]
Area	highway 5 km
Number of lanes	3
Percentage of accident	20%
Default threshold	0.0
Threshold increment	0.05
Propagation delay (ms)	1
Data rate (Mbps)	6

TABLE I
Simulation Parameters

VI. Simulation Results and Analysis

Fig. 9 gives an example of a vehicle in a scenario with a neighborhood density of 10 veh/km/lane and 20% of attackers. The crossed line shows the evolution of the threshold during the simulation. The solid line represents the ratio (i.e. the observation function ξ) which increases when the vehicle approaches the event location (figured by vertical dashed line for real warning and vertical dotted line for fake warning). In this example, at 130 sec, the vehicle receives a series of fake warnings that quickly increase the threshold and lead to a stable threshold value. Indeed, in these simulations, the threshold is stable at 40% of the current neighborhood. We conclude that it overestimates the current level of attackers.

Another important aspect is the threshold reduction. Indeed, if attackers are localized in a specific location, then they will impact the threshold for the entire journey of a vehicle that goes through this location. Therefore, the vehicle will be

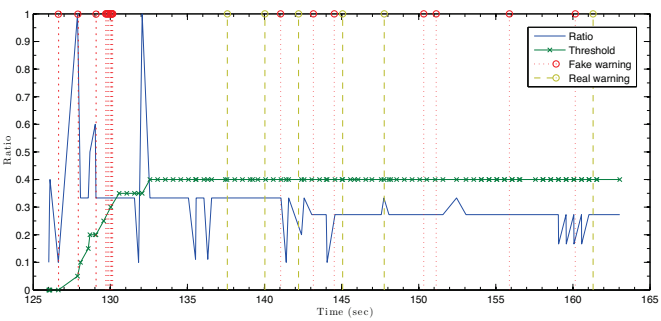

Fig. 9. Evolution of ratio and threshold

suspicious even if there is no more attackers. An approach could be that a series of good decisions during a certain delay reduces the threshold. We define as *good decision* a decision made where the vehicle does not react to a fake warning and reacts to a real warning. This issue will be considered in our future work.

Fig. 10 shows the average decision delay in function of the density for a percentage of attackers in the interval [0%, 50%]. Having 50% of attackers is impossible with the assumption of a majority of honest nodes but permits to show the worst case. The decision delay is lower than 5 seconds and is independent from the percentage of attackers. The high value at 50% for 10 veh/km/lane is not representative because the vehicle only takes on average three reactions during these simulations.

Fig. 11 shows the percentage of wrong decisions in function of the percentage of attackers for a density of 5 veh/km/lane and 10 veh/km/lane. We remark that with a percentage of attackers greater than 10% the vehicle makes a wrong decision with a percentage above 50%. Moreover, with a low density of 5 veh/km/lane, the threshold takes time to be stable and so does not avoid the information forgery attack. With a higher density, the threshold adapts faster and avoids a 100% of wrong decisions. From Fig. 11 we conclude that the simulation time needs to be increased to analyze the delay needed to reach a stable threshold (because this delay is strongly dependent of the scenario), and to increase the number of suffered events per vehicle. Indeed, contrary to Fig. 9, Fig. 11 shows an average of all simulations, and thus, does not provide a stable threshold value. Aiming for a stable threshold is relevant in our simulations because we assume an uniform percentage of attackers. But, in more realistic scenarios, this percentage will vary, and thus, the stability of the threshold is not relevant.

We also conclude that starting with a threshold equal to 0 suffers from a long bootstrapping phase, so the vehicle reacts to fake warnings during this phase. To alleviate this issue, we could set the threshold to a higher default value. Therefore, we could consider the reputation score or the trust score that could be computed thanks to the six sources as proposed in [10]. Sociological studies that estimate the potential percentage of attacker of the system could also be taken into account in the default threshold.

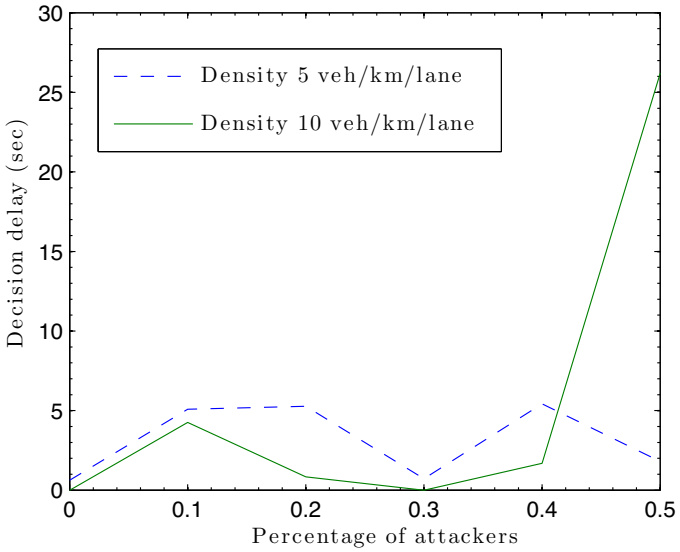

Fig. 10. Average decision delay in function of percentage of attackers

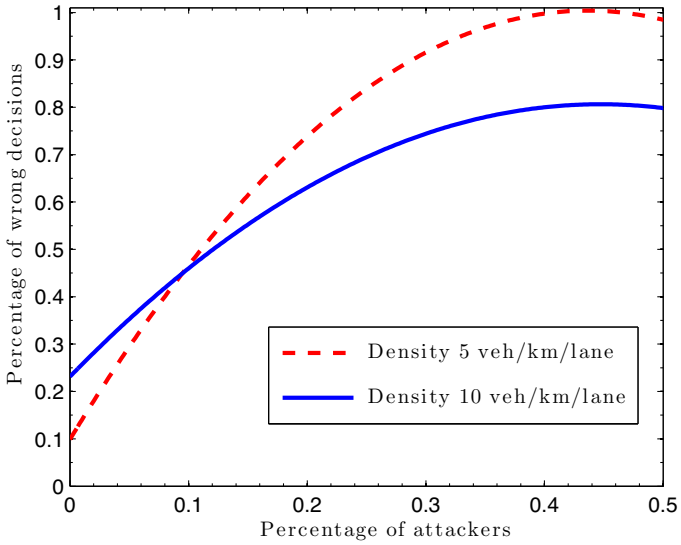

Fig. 11. Percentage of wrong decisions

VII. Conclusion

In this paper, we investigated the problem of threshold establishment in VANET. We proposed a dynamic threshold mechanism to increase trust in local danger warning by detecting spoofed data. More specifically, we modeled the threshold as a Kalman filter. We proposed an algorithm similar to a learning scheme to dynamically adjust this threshold. Thus, the threshold estimates the current percentage of attackers in the VANET. We provided simulations and analyzed the impact of the density and the percentage of attackers on the decision delay and the percentage of wrong decisions. Our method overestimates the presence of attackers but leads to protect vehicle from spoofed data injection. We conclude that the default threshold value should be chosen carefully to shorten the inevitable bootstrapping phase. Currently, we are working on further extensive simulations to assess the delay to achieve a best-suited threshold.

Acknowledgment

This work has been partly done in the framework of COST IC0906 WiNeMO "Wireless Networking for Moving Objects (2010-2014)" project during the stay of Jonathan Petit at Tampere University of Technology. Authors would like to thank Olga Galinina and Sergey Andreev for their comments. The research leading to these results has also received funding from the European Union's Seventh Framework Programme project PRESERVE under grant agreement n°269994.

REFERENCES

[1] P. Papadimitratos, F. Kargl, M. Weber, and T. Leinmuller, "Secure vehicular communications: Design and architecture," *IEEE Communications Magazine*, vol. 46, no. 11, pp. 100–109, November 2008.

[2] IEEE, "Trial-use standard for wireless access in vehicular environments - security services for applications and management messages," *IEEE Standard 1609.2-2006*, 2006.

[3] M. Raya, "Data-centric trust in ephemeral networks," *PhD thesis*, June 2009.

[4] L. Lamport, R. Shostak, and M. Pease, "The byzantine generals problem," *ACM Trans. Program. Lang. Syst.*, vol. 4, pp. 382–401, July 1982.

[5] M. Fischer, "The consensus problem in unreliable distributed systems (a brief survey)," in *Proceedings of the 4th International Conference on Fundamentals of Computation Theory (FCT'83)*, August 1983, pp. 127–140.

[6] Z. Cao, J. Kong, U. Lee, M. Gerla, and Z. Chen, "Proof-of-relevance: Filtering false data via authentic consensus in vehicle ad-hoc networks," in *IEEE INFOCOM Workshops 2008*, Phoenix, AZ, USA, April 2008, pp. 1–6.

[7] J. Petit, "Analysis of ecdsa authentication processing in vanets," in *Proceedings of the 3rd international conference on New technologies, mobility and security (NTMS'09)*, Cairo, Egypt, 2009, pp. 388–392.

[8] B. H. Kantowitz, R. J. Hanowski, and S. C. Kantowitz, "Driver acceptance of unreliable traffic information in familiar and unfamiliar settings," *Human Factors*, vol. 39, no. 2, pp. 164–176, 1997.

[9] B. Ostermaier, F. Dötzer, and M. Strassberger, "Enhancing the security of local danger warnings in vanets - a simulative analysis of voting schemes," in *Proceedings of the 2nd International Conference on Availability, Reliability and Security (ARES'07)*, Vienna, Austria, April 2007, pp. 422–431.

[10] 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 7th ACM international workshop on Vehicular interNETworking (VANET '10)*, 2010, pp. 51–60.

[11] T. Leinmüller, E. Schoch, F. Kargl, and C. Maihöfer, "Decentralized position verification in geographic ad hoc routing," *Wiley Security and Communication Networks Journal*, vol. 3, no. 4, July 2010.

[12] S. Dietzel, E. Schoch, B. Konings, M. Weber, and F. Kargl, "Resilient secure aggregation for vehicular networks," *IEEE Network*, vol. 24, no. 1, pp. 26–31, January 2010.

[13] T. Kosch, "Local danger warning based on vehicle ad-hoc networks: Prototype and simulation," in *Proceedings of the 1st International Workshop on Intelligent Transportation (WIT'04)*, Hamburg, Germany, March 2004, pp. 43–47.

[14] J. Mittag, F. Thomas, J. Härri, and H. Hartenstein, "A comparison of single- and multi-hop beaconing in vanets," in *Proceedings of the 6th ACM international workshop on VehiculAr InterNETworking (VANET'09)*, Beijing, China, 2009, pp. 69–78.

[15] E. Schoch, F. Kargl, M. Weber, and T. Leinmuller, "Communication patterns in vanets," *IEEE Communications Magazine*, vol. 46, no. 11, pp. 119–125, November 2008.

[16] J. Petit and Z. Mammeri, "Dynamic consensus for secured vehicular ad hoc networks," in *Proceedings of the 7th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob'11)*, Shanghai, China, October 2011, pp. 1–8.

[17] F. Kargl, M. Zhendong, and E. Schoch, "Security engineering for vanets," in *Proceedings of the 4th Workshop on Embedded Security in Cars (ESCAR'06)*, Berlin, Germany, 2006.

[18] L. Wischhof, A. Ebner, H. Rohling, M. Lott, and R. Halfmann, "Self-organizing traffic information system," *Inter-Vehicle-Communications Based on Ad Hoc Networking Principles : The Fleetnet Project*, pp. 233–272, 2005.

[19] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation*. Prentice Hall information and system sciences series, 2000.

[20] P. S. Maybeck, *Stochastic Models, Estimation, and Control: Volume 1*. Academic Press, 1979.

[21] "The Network Simulator NS-2," http://www.isi.edu/nsnam/ns/.