Reinforcing Traffic Safety by Using CAM to Verify Velocity Accuracy

Erik de Britto e Silva*†, Seilendria A. Hadiwardoyo†, Cristina Emilia Costa* and Johann M. Marquez-Barja†

*Smart Networks and Services, Fondazione Bruno Kessler, Trento, Italy

E-mail: {edebrittoesilva,ccosta}@fbk.eu

†University of Antwerp - imec, IDLab research group, Antwerp, Belgium

E-mail: {erik.debrittoesilva,seilendria.hadiwardoyo,johann.marquez-barja}@uantwerpen.be

***Abstract*—The benefits of Cooperative Intelligent Transport Systems (C-ITS) span a range of areas, such as road safety and service reliability. Thus, traffic accidents can be avoided, and ultimately human lives can be saved. As C-ITS disseminate in-vehicle information, an error in the disseminated velocity accuracy may cause accidents, particularly in the automated and autonomous driving contexts. Hence, it is vital to detect these errors and warn vehicles about the inaccuracy of the speed reported by the vehicle in the C-ITS Cooperative Awareness Message (CAM). To illustrate the potential of our solution, we present one use case with platooning green light priority and another to save the life of a cyclist when it jumps into the lane of a vehicle without signaling or without enough space to do so. Our solution integrates a Roadside Unit (RSU) with a velocity detection device to enforce the accuracy of the disseminated velocity from a vehicle using C-ITS. We compare the velocity disseminated by a vehicle, via CAMs, with the velocity acquired from a standard deployed "speed detection device". To show the feasibility of our proposal, we emulate a vehicle sending and receiving C-ITS messages in a virtualized environment. The RSU receives C-ITS messages to get the disseminated velocity of a vehicle and, if needed, warn that the disseminated velocity is inaccurate. Focusing on our experiments, it takes less than 500μs to process the information and report the inaccuracy. Additionally, our work introduces a data calibration warning that could be needed by autonomous vehicles.**

***Index Terms*—C-ITS, CAM, ITS, RSU, V2X, vehicular communications, velocity accuracy.**

I. Introduction

Cooperative Intelligent Transport Systems (C-ITS) are systems evolved from Intelligent Transport Systems (ITS), both defined by European Telecommunications Standards Institute (ETSI) standards. C-ITS enable vehicles to interact directly with each other and with the surrounding road infrastructure, leveraging the ITS ecosystem. In road transport, C-ITS typically involves communication between vehicles - Vehicle-to-Vehicle (V2V), between vehicles and infrastructure - Vehicle-to-Infrastructure (V2I) and or Infrastructure-to-Infrastructure (I2I), and between vehicles and pedestrians or cyclists - Vehicle-to-Pedestrians (V2P). V2P comprises people walking, children being pushed in strollers, people using wheelchairs or other mobility devices, passengers embarking and disembarking buses and trains, and people riding bicycles. C-ITS benefits span a wide range of areas, improving road safety, reducing travel times, optimizing transport efficiency, enhancing mobility, increasing service reliability, reducing energy use and CO_2 and pollutant emissions, and ultimately reducing road fatalities.

ETSI defines a set of messages for C-ITS communication between V2V and V2I, the so-called Cooperative Awareness Message (CAM) [1], which disseminates information about the vehicle type, position, and dynamics; and the Decentralized Environmental Notification Message (DENM) [2] disseminates events or conditions on the road. The Roadside Unit (RSU) is a communication station near the road that communicates with the road authority and is responsible for sending and receiving messages to vehicles.

There is no standardized solution to check if a CAM content is accurate or not. Vehicles can send untrustworthy data due to malfunction, a faulty sensor, or intentionally altered. We pose the following unanswered question: if a speed sensor delivers a wrong measurement, how can it be detected or isolated in a cooperative environment? When the system has no way to detect this error, an external warning is needed. Each CAM message contains the velocity of the vehicle, thus both speed and direction vector are disseminated. Unfortunately, C-ITS has not yet defined a way to check the received velocity values and to deliver warning messages about any data inaccuracy.

We propose to detect the accuracy of the disseminated velocity and to warn vehicles, and a central authority when it is inaccurate by using CAM messages sent from the vehicle over the ITS-G5 access layer technology [3]. With this feature, we support the vehicle to keep the accuracy of its informed velocity. It works in an opportunistic way when the vehicle is scanned by a velocity detection device embedded in an RSU, as shown in Figure 1. In this way, the velocity of the vehicle is sampled by two devices, i.e., (1) the velocity delivered by the CAM from the speedometer of the vehicle, and (2) the detected velocity by the RSU velocity detector. Then, the velocity value sampled by the in-vehicle system is compared with the value sampled by the velocity detection device located at the RSU. Depending on the difference, a warning DENM message is sent to the vehicle.

Our main motivation is that accidents can be avoided by detecting and notifying inaccuracies of the velocity disseminated among vehicles, and reducing fatalities because. Inaccurate velocities increase the risk of road fatalities due to miscalculated

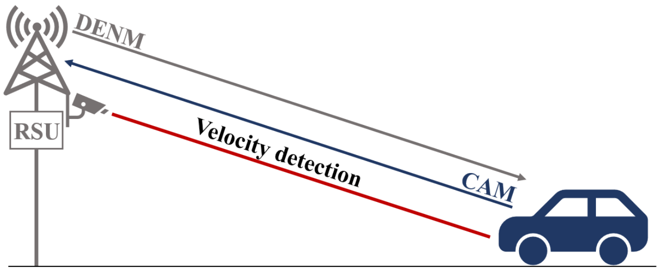

Fig. 1: Velocity detection and C-ITS messages

distances. Moreover, our work closes a loophole to be used by attacking hackers using fake data. Velocity is important information for many existing safety applications, mainly for the two following warning applications: (1) **Intersection Collision Risk Warning (ICRW)** [4]; and (2) **Longitudinal Collision Risk Warning (LCRW)** [5].

In this work, we propose combining the use of CAM and DENM messages and C-ITS infrastructure (speed-cameras and RSUs) to check the accuracy of a disseminated velocity value of a vehicle, and to issue a warning message in case of inaccuracy, with a negligible latency within a C-ITS environment. To study the feasibility of our work, we present a testing scenario comprised of an RSU capable of sampling the velocity of a passing vehicle in a virtualized environment. The RSU compares the velocity sent from the vehicle with the sampled by the RSU to check the accuracy.

The importance of working with accurate measures is presented in the "Vehicle and RSU data calibration" use case C.3.19, in the ETSI Technical Report 102 638 [6], to provide improved service quality and improved maintenance management for vehicles and RSUs through online calibration. This use case presents an application in which an RSU compares its sensed data, or the calculated traffic conditions, to the respective data disseminated from passing vehicles, using V2I communication. Our proposal is an implementation of the aforementioned use case to enforce the accuracy of the velocity of a vehicle and to the most important goal: to prevent fatal accidents. As an added benefit, autonomous vehicles could need our data calibration warning.

II. Related Work

To the best of our knowledge, our work is one of the first studies in checking the velocity value accuracy of a C-ITS CAM using a velocity detection device. Since these devices are frequently checked and calibrated according to road regulations, they can be used as an accurate tool. Our idea is similar to an IoT watchdog [7], an application that monitors and detects anomalous behavior of IoT devices, typically caused by faults or attackers. After detecting the anomalous behavior, the watchdog commands the IoT device to a reboot or firmware install. In our case, the watchdog sends a warning C-ITS message about an inaccurate velocity value.

In [7], authors deploy a cooperative watchdog system to detect misbehavior nodes [7] in Vehicular delay-tolerant networks (VDTN). VDTN have a different protocol for sharing data among the mobile nodes (vehicles) with store-carry-and-forward bundles. The misbehavior node is defined as a node not following the protocol, which degrades the network performance. In our work, based on C-ITS, the CAM message is sent at maximum in one message per second (1Hz) frequency, and the misbehavior node is a vehicle that disseminates inaccurate velocity to its neighbours by sending CAMs. Our solution warns C-ITS neighboring vehicles and roadside units by sending DENMs.

In an effort aiming to provide a state-of-art of misbehavior detection mechanisms in C-ITS [8], the authors present two types of attacks common and relevant for C-ITS, the Sybil attack and the bogus information attack. Some mechanisms and their feasibility concerning standards and law compliance are analyzed. The requirements in software and hardware of each one are also investigated. In the Sybil attack, the attacker could take advantage of pseudonyms to launch a kind of bogus information attack. In the bogus information attack, the attacker sends bogus data from apparently present vehicles. In our case, an attacker can send bogus velocity parameters, but the bogus values are detected, and warnings are disseminated. A pseudonym or pseudonym certificate is a unique individual certificate used as a temporal identifier of a vehicle to protect privacy, generated each time depending on some occurring conditions. For example, a vehicle must change its pseudonym every 500m traveled [9].

Towards the analysis of different Machine Learning (ML) based solutions, authors in [10] propose a simulation work based on local misbehavior detection information sent by the Onboard Unit (OBU) like our case, and by RSUs, called Misbehavior Reports (MBR), to a Misbehavior Authority (MA). The work adds the reporting functionality to OBUs and RSUs, requires much processing power and storage space. The goal is to evaluate different ML approaches for the MA, and it is not adequate for easy standardization and easy implementation. Our proposal is simple and easily implemented with a velocity detection device directly embedded in an RSU. It detects misbehavior in the disseminated velocity based on its measured accuracy by an RSU.

III. Background

A. *In-vehicle measured velocity*

Using the velocity provided by a C-ITS public disseminated CAM without modifying the standard is the first benefit of our proposal. According to Regulation Number 39 of the Economic Commission for Europe of the United Nations (UN/ECE) [11], the Velocity Indicated (VI) by a vehicle cannot be less than its True Velocity (VT). With the following relationship to ΔV, calculated as VI - VT:

$$0 \leq \Delta V \leq 0.1VT + 4km/h \quad (1)$$

Therefore, the VI sent by each vehicle less the VT is required to provide values not greater than the above mentioned in (1), namely less or equal than the 10% of the true velocity, plus 4 km/h. Otherwise, the vehicle has an incorrect functioning.

B. Detection device

As an example of velocity detection equipment, we assume using the RADARXENSE RXS-DR-10 equipment [12] shown in Figure 2, with its specifications shown in Table I. The velocity detection range is from -70m/s, for outcoming vehicles, to +70m/s for incoming vehicles, 0.5% accuracy, 0.3m to 100m detection range distance, and 10ms readout per sample. The detection range distance is defined as the range between the minimum and maximum distances when a passing vehicle has its velocity correctly sampled by the detection device.

Fig. 2: RADARXENSE RXS-DFR-10 [12]

TABLE I: RXS-DFR-10 Specification

Velocity range	-70m/s to +70m/s (252km/h)
Minimum velocity	10cm/s (0.36km/h)
Accuracy	0.5%
Detection range distance	0.3m to 100m
Readout time	10ms

C. Velocity detection tolerance margin

The VI delivered by the CAM, as required by C-ITS, has an accuracy magnitude lesser than the accuracy provided by the velocity detection device, more than 10% (as shown in section III-A) versus 0.5% (Table I). We consider the accuracy of the detection device as sufficient to detect the VI inaccuracy, as it is 20 times greater than the accuracy of the velocity from a vehicle speedometer (0.5% divided by 10%). We adopted the Belgian regulations [13] to deploy a realistic velocity tolerance margin. We consider the regulations for cars or motorcycles weighing up to 3.5 tons of mass, which define a tolerance margin divided into two ranges of velocity, as shown in Table II. The velocity tolerance margin is the maximum value that a VT can be over the velocity limit but still considered as under it by the road regulations.

The tolerance margin is always a positive value. For our goals, we consider the velocity to be inaccurate in two cases: (1) when VI minus VT is greater than the value of the tolerance margin, and (2) VI is lesser than VT by any detected difference. The ideal situation would be inaccuracy lesser than 1%, that will result in negligible miscalculations in distances to prevent collisions or warn about collision risks. When a vehicle passes in front of a velocity detection device and sends a CAM message to the RSU, we compare the velocity value delivered by the CAM with the detected velocity. If the velocity is checked to be over the tolerance margin or it is under the detected velocity, a warning message is sent by the RSU to the vehicle and its neighbors. The message is a new type of DENM.

TABLE II: Tolerance margin VI $\geq$ VT [13]

Velocity	Margin
$\leq$ 100km/s	6km/h
$>$ 100km/s	6%

D. Latency

1) CAM RTT: We must estimate the Round Trip Time (RTT) of the CAM to perform the comparison. If it is too high, the comparison can be invalidated. It must be low enough to consider the two velocity values taken simultaneously for the comparison.

A work by Große et al. [14] evaluated the RTT between two ITS-G5 stations, and found out that the RTT is $\approx 500\mu$s; it is important to highlight that this RTT includes the times needed to: (1) assemble, and transmit the packet by the sender, plus (2) receive this packet, assemble a response packet, and transmit this packet by the receiver (as quick as possible); and finally (3) receive, and process the response back at the sender. We consider each CAM RTT to have a negligible impact on the latency of the messages. As each CAM, as shown in Section VI, needs to be transmitted at a maximum of 100ms, the 500μs delay is two orders of magnitude less than the time between two transmitted CAMs. Our solution adds a negligible delay.

2) CAM generation from a vehicle: ETSI European Norm 302 637-2 [1] defines how a CAM is generated, and five time parameters. The time parameters are:

1) TMAX = 1000ms $\rightarrow$ maximum time between CAMs
2) TMIN = 100ms $\rightarrow$ minimum time between CAMs
3) TAPP $\rightarrow$ maximum time required by a C-ITS application to send the next CAM
4) TDCC $\rightarrow$ minimum time required by the Decentralized Congestion Control (DCC) to send the next CAM
5) T is the elapsed time since the last generated CAM

When the engine of a vehicle starts, it generates the first CAM. Then a CAM is sent periodically like a beacon from TMIN to TMAX, but other conditions are involved that can change the time to send the next CAM. Suppose an application needs to send the following message in less than TMIN. In that case, the parameter TAPP is set lower than TMIN, e.g., for sending a time-critical message to avoid congestion of the communication traffic among vehicles. The DCC sets the TDCC to have a higher value than TMIN depending on the packet collision rate. Higher packet collision rate results in higher TDDC for decreasing the packet collision rate. As the packet collision rate drops, the TDCC drops to lower values. Depending on the following three absolute thresholds, a vehicle must generate a CAM before T reaches TMIN or TDCC:

1) $|\Delta Heading| > 4^\circ \rightarrow$ its heading variation is greater than 4°

2) $|\Delta Movement| > 4m \rightarrow$ it moved more than 4m
3) $|\Delta Speed| > 0.5m/s \rightarrow$ its speed variation is greater than 0.5m/s

The flowchart depicted in Figure 3 presents how to generate a CAM according to the ETSI definitions. After the initial CAM, T is set to zero. While T is lesser than TMIN, it sleeps by an undefined time TS until T is greater than or equal to TMIN, then T is checked if it is also greater than or equal to TDCC. In this case, a CAM must be generated. Otherwise, the three thresholds are checked. If at least one of the thresholds is met, a CAM must be generated. Otherwise, if T is greater than or equal to TMAX or greater than or equal to TAPP, a CAM must be generated. While T is still not greater than or equal to TMIN and T is not greater than or equal to TDCC, it sleeps for an undefined time TS until the value of T is greater than or equal to TMIN. Then it returns to the first condition at the flowchart, checking if it is also greater than or equal to TDCC. Internal timers and events activate the condition cheking.

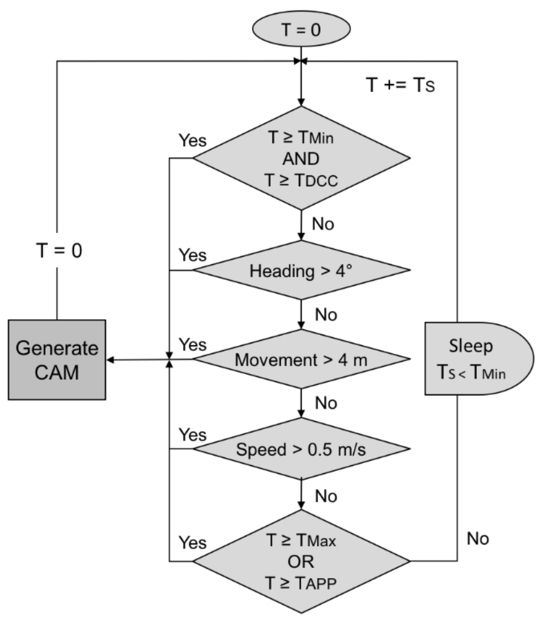

Fig. 3: CAM generation from a vehicle [15]

We present in Table III how the CAM interval decreases with the increase of the velocity, as for each $|\Delta Movement| > 4m$ a CAM must be generated, meaning an increase on the frequency to send CAMs. We provided velocity values of a vehicle from 14.4 km/h to 144 km/h, providing from one CAM per second to 10 CAMs per second, namely the minimum and maximum CAM intervals, respectively. Therefore, moving vehicles with a velocity greater or equal to 144km/h will generate a new CAM message every 100ms, and moving vehicles with a velocity less or equals to 14.4km/h will generate a new CAM per second, as limited by the ETSI standard. These limits will apply to vehicles moving in a straight line along a lane, under normal conditions.

3) Detection device sampling interval: There is a range of 100 meters to detect the velocity of a vehicle at each lane, in the direction of the velocity detection equipment, as shown in Table I. The RXS-DFR-10 [12] device generates messages at every 10ms (100Hz), faster than the fastest CAM interval (10Hz), for a vehicle moving in a straight line in a lane. Table IV shows how many velocity detection samples are taken during each CAM sending interval, depending on the velocity of the vehicle presented in Table III.

TABLE III: CAM frequency increase with velocity

Velocity		CAM interval	Frequency
km/h	m/s	ms	Hz
14.400	4.000	1000.000	1.000
30.000	8.333	480.192	2.082
60.000	16.667	239.995	4.167
90.000	25.000	160.000	6.250
120.000	33.333	120.001	8.333
144.000	40.000	100.000	10.000

TABLE IV: Velocity detection samples

Velocity (km/h)	CAM interval (ms)	Samples in 10ms
14.400	1000.000	100
30.000	480.192	48
60.000	239.995	23
90.000	160.000	16
120.000	120.001	12
144.000	100.000	10

The detection device targets a static point in the middle of a fixed imaginary rectangle area over the road to sample the speed of a vehicle in a determined lane. The rectangle has its length greater around 4 meters along the road. Since a moving vehicle reaches the far edge of the rectangle until it reaches the near edge to the detection device at least one CAM is sent because the vehicle has reached $|\Delta Movement| > 4m$. The received CAM is compared with the current velocity sample.

E. Privacy

The use of speed detection equipment is regulated by law [16]. All the existing process of taking pictures of vehicles and identifying the license plate happens in strict compliance with the regulations. It is also submitted to frequent auditing. The procedure to disseminate the inaccuracy of the velocity of a vehicle to the vehicle itself and to its neighbors using DENM messages can be done from a technical point of view. However, depending on local regulations, such procedure may not feasible or realizable. We consider that warning vehicles about an inaccurate speed is a public safety matter. It avoids traffic accidents due to miscalculated distances. Moreover, in the case of autonomous vehicles such notification will enable calibrating their velocity values. ETSI standards for CAM and DENM [2] also deploy temporal Station Identifiers or pseudonyms, already explained in section II, which protects the privacy of each vehicle. There is a velocity detection equipment from Jenoptiks [16] that augments the characteristics of the velocity detection. It is capable of multi-lane and multi-target measurements, providing for each vehicle: (1) velocity, (2) picture of the driver, (3) vehicle type or class,

(4) license plate number, (5) lane, (6) date and time, and capable of recording live videos of the road traffic for road authority purposes according local regulations, as shown in Figure 4. Therefore, we can use CAM position information to check with the vehicle position detected by the equipment and then compare the velocity value acquired from both. It occurs when the position information of the vehicle, given by the Global Positioning System (GPS) coordinates, is near the fixed position targeted by the detection device at the road.

Fig. 4: Jenoptiks's snapshot of a real road traffic live Video - [16]

IV. Pontential use cases

In this section, we present two potential use cases affected by miscalculated distances caused by the dissemination of inaccurate velocities.

A. *Platooning green light priority*

Our first potential use case is a platooning of trucks concentrating on the Leading Vehicle (LV). According to the implementation from S. Ellwanger and E. Wohlfarth [17], with real-world values, trucks keep a distance of 15m among each other and a velocity of 80km/h. The expected time between two consecutive CAMs is 0.1s. They also deploy sensors to avoid the risks of relying only on C-ITS information to keep truck distances. The received data from C-ITS messages are compared with sensors data. We envision an application in the infrastructure to prioritize the platoon to pass by green traffic lights. An RSU receives the length of the platoon, the position and velocity of the LV, and calculates when and how long to switch traffic lights to green for passaging the platoon. If the LV disseminates an inaccurate velocity, two errors could happen. Either the value is over or under the real velocity.

When the velocity of the LV is under the real value, it causes the traffic lights to switch to green earlier than would be required. The traffic lights will also wait less time in green before the last vehicles pass. The platooning will break into two parts and will stop the last vehicles of the platooning. It results in two undesirable effects: it wastes green light time before the arrival of the LV and breaks the platooning.

When the velocity of the LV is over the real value it causes the traffic light to stay more time in red, making the platooning stop and wait for the green light. It results in one undesirable effect that is to stop the platooning.

Our solution to assure velocity data accuracy will cause the RSU to send DENMs with the detected inaccuracy. The road infrastructure will calculate the correct time to switch the traffic lights to green and the correct time to wait for the last vehicle to pass the green light.

B. *Cyclist jumping into a lane of a vehicle without signalling*

Our second potential use case is between a Vulnerable Road User (VRU)[1] and a vehicle. The VRU considered in this case is a cyclist. The cyclist and the vehicle are C-ITS enabled entities. Suppose that the cyclist is abruptly jumping into a lane of a vehicle on the road. As the cyclist approaches the vehicle, they will notice each other by a C-ITS application. But, depending on the inaccuracy of the velocity of the vehicle, say the vehicle disseminating and using an erroneous velocity value, an accident will have a high probability to happen. Our solution provides the correct values with the potential to avoid an accident that could cause the death of the cyclist.

Between 2007 and 2016, 21787 cyclists died from road accidents in the European Union, as published by European Road Safety Observatory (ERSO) [18]. Undoubtedly, lives will be saved by enabling cyclists with C-ITS equipment and implementing our solution. Moreover, if we extend our solution to pedestrians, more lives will be saved as between 2015 and 2017, 99% of all pedestrian deaths were caused by an impact with a motor vehicle [19].

V. Experimental Setup

To show the feasibility of our proposal, we implement an environment comprising two entities with applications to run our experiments in an emulated vehicular environment in a virtual machine. Next sections present the environment, and methodology for our experiments.

A. *Experimental Environment*

The environment for our experiments, as shown in Figure 5, comprises the following entities:

1) Roadside unit: A C-ITS entity running applications to receive CAM messages with the velocity from the vehicle, detect the vehicle velocity using a speed detection device, compare the two velocities, and launch an application to send DENMs, when the difference between the velocities exceeds the tolerance margin
2) Vehicle: A C-ITS entity running one application to send CAM messages, with information about the vehicle velocity, to the RSU, and another application to receive DENM messages warnings from the RSU. The dedicated device for V2X communications embedded in the vehicle is called OBU. A GPS emulator provides the points along a predetermined trajectory moving at a constant velocity

[1]https://ec.Europa.eu/transport/themes/its/road/action_plan/its_and_vulnerable_road_users_en

B. Methodology

We deploy the open-source implementation of the ETSI C-ITS protocol stack Vanetza[2] project [20] as the framework for our experiments, using the GeoNetworking (GN) protocol [21] over Ethernet frames as the access technology. The path of a moving vehicle is emulated in the framework using a GPS emulator; simultaneously, the vehicle sends GN packets that are broadcasted among surrounding vehicles and RSUs using the same access technology. Each application receives its GPS coordinates generated by the GPS emulator. A previously generated file using Google Maps[3] contains the vehicle movement, converted in National Marine Electronics Association (NMEA)[4] messages, feeds the GPS emulator. The GN protocol is used to broadcast packets over a geographical area.This way, C-ITS messages are broadcasted between the connected entities of the experiments: the RSU and the vehicle. The environment runs in a virtualized Linux machine using Oracle VM VirtualBox™. As shown in Figure 5, a CAM sent by the vehicle and received by the RSU comprises two separated Vanetza applications, one to encode and send the CAM by the vehicle, and the other to receive and decode the CAM by the RSU. Similarly, a Vanetza application encodes and sends a DENM by the RSU, and other receives and decodes the DENM by the OBU. We do not use the security layer of the ETSI C-ITS protocol.

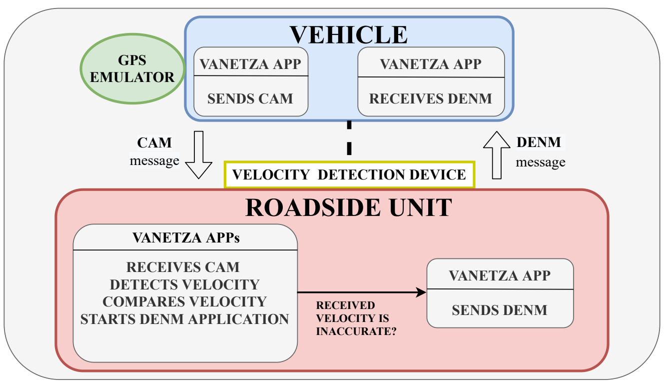

Fig. 5: Entities, Vanetza applications, velocity detection, and messages

The movement of the vehicle is over a straight line of about 200m, drawn in a street of the city of Antwerp - Belgium.

The experiment starts when the vehicle uses its application to send CAM messages and emulates its movement according to the coordinates from the GPS emulator at a constant velocity. Simultaneously, the RSU application starts waiting to receive the first CAM of the vehicle. The UML sequence diagram presented in Figure 6 shows the experiment flow, where the RSU and the vehicle OBU are receiving and sending CAM messages, respectively.

[2]https://www.vanetza.org
[3]https://maps.google.com
[4]https://www.nmea.org

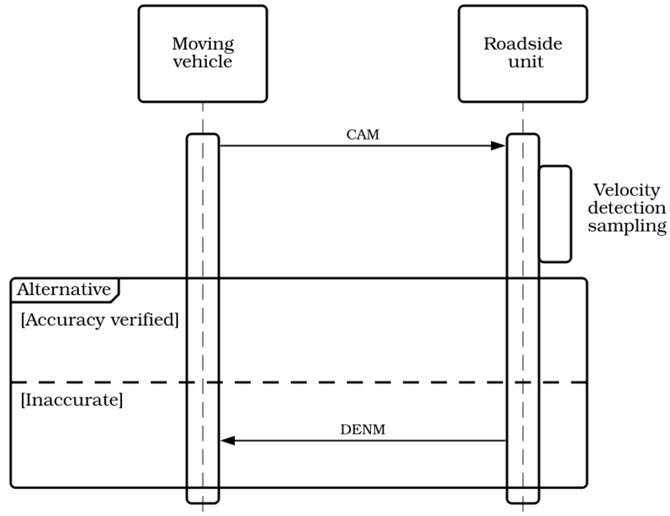

Fig. 6: Experiment flow

VI. Results

Our experiments emulate two cases with three different velocity accuracy delivered from CAM messages sent by a vehicle. The first case covers detected velocities values under or equals to 100km/s, we chose 90km/h. Parameters used in this experimentation are shown in Table V. The second case covers detected velocities values over 100km/h, we choose 180km/h, with their corresponding values shown in Table VI. The tolerance margin adopted is shown in Table II, for the first case (velocity $\leq$ 100km/h) is 6km/h, and 6% for the second case (velocity $>$ 100km/h).

The conditions of each velocity accuracy are:

1) Disseminated value below the detected speed: a DENM is sent to the vehicle to warn this inaccuracy condition
2) Accurate velocity: no notification is required nor sent
3) Disseminate value above the tolerance margin: a DENM is sent to the vehicle to warn this inaccyracy condition

TABLE V: Detected velocity $\leq$ 100km/h

CAM velocity (km/h)		Detected velocity (km/h)
Below detected velocity	80	90
Accurate velocity	95	90
Above detected velocity and the tolerance margin	120	90

TABLE VI: Detected velocity $>$ 100km/h

CAM velocity (km/h)		Detected velocity (km/h)
Below detected velocity	150	180
Accurate velocity	189	180
Above detected velocity and the tolerance margin	198	180

Figures 7 and 8 show the measured processing time of the three conditions for the first and second cases. Each condition was repeated for 33 rounds. We define processing time as the time since the RSU receives the CAM until it sends the DENM. It follows the steps:

1) The moving vehicle sends CAMs as it passes by the RSU
2) The RSU senses the passing vehicle, samples the velocity, and records the received CAM
3) A RSU application calculates the accuracy of the received velocity
4) If velocity is inaccurate, the RSU sends a DENM to warn about the inaccurate velocity
5) The vehicle receives the DENM

As the processing time of our experiments is two orders of magnitude lesser than the 10ms latency of the velocity detection device, we show that our proposal adds a negligible delay to existing equipment, as shown in Figures 7 and 8. High processing time will delay the DENM message to be received by a vehicle and its neighbors; depending on the delay value, it could cause fatalities. High processing time will provoke higher miscalculations until the inaccurate velocity warning is received, and the velocity value is removed/banned from the vehicles as invalid. It has a potential impact on regulation and legislation to stop or remove vehicles from the road due to inaccurate velocity. This way, the DENM message issued to C-ITS enabled vehicles to warn of inaccurate velocity values, is useful for fatalities prevention.

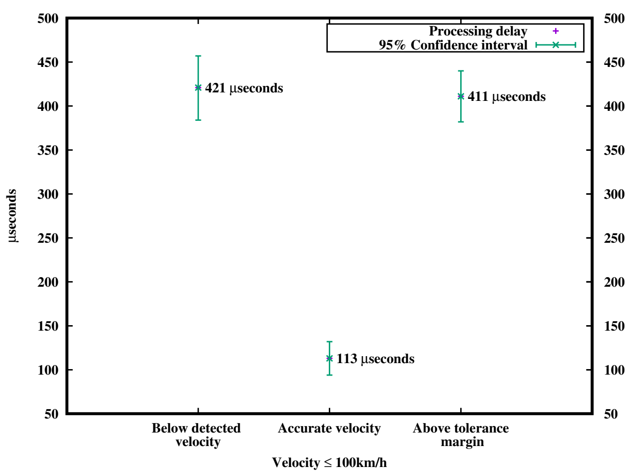

Fig. 7: First case measurements of the processing delay

Our experiments performes in less the 500μs, the same magnitude as the message RTT, as presented in Section III-D1. As CAM and DENM disseminates in the minimum time of 50ms, as defined by the parameter T_{OFF} at Table A.2 in ETSI TS 102 687 [22], this performance ensures our proposal suitability to be deployed in a real C-ITS environment. The suitability is kept using a real velocity detection device [12] with a readout time of 10ms, which is only 20% of the T_{OFF}.

Our solution scales up to 6 lanes in each direction in a highway with 4m lane width on a 5km long straight road under certain conditions; according to a published work from Shimizu et al. [23]. The authors showed the feasibility of a vehicle density of 108 vehicles/km moving at 100km/h,

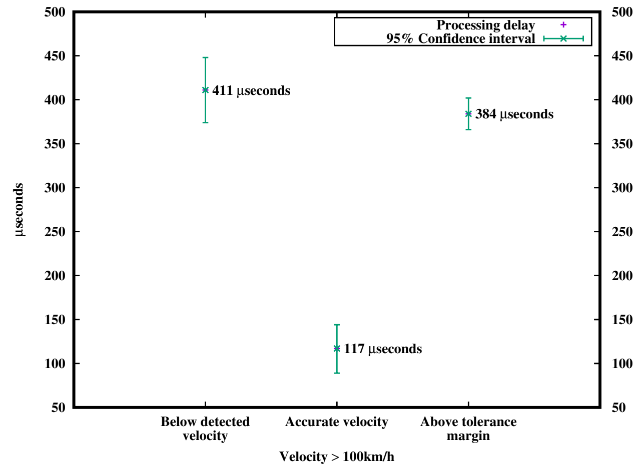

Fig. 8: Second case measurements of the processing delay

separated for a distance of 4s (27.78m). Each vehicle sends one CAM per 10 milliseconds, it achieves over 90% packet received rate for distances lesser than or equal to 100m.

VII. Conclusion

In this work, we propose a low-complexity solution to avoid accidents on the roads, putting together the C-ITS environment facilities and existing velocity detection devices, already operating under local regulations. A vehicle disseminating an accurate velocity is an important feature to surrounding vehicles and to the traffic authority. C-ITS entities need the real velocity to make the right calculations about distances to avoid the risk of fatalities. When a vehicle disseminates inaccurate velocities, we consider two different cases: (1) under the detected velocity, and (2) over a tolerance margin.

There is no need for high-complexity hardware neither algorithms in deploying our solution. We show two cases, based on velocity ranges, to check the accuracy of a disseminated velocity (the same value from the vehicle speedometer) using a C-ITS CAM message, compared with the velocity sampled by a velocity detection device. We measure the emulated delay added by the processing time of our solution in three conditions: (1) when the velocity of a vehicle is below the disseminated velocity, (2) when the velocity of a vehicle is covered by a tolerance margin (accurate), and (3) when the velocity of a vehicle is above the tolerance margin. In our emulated environment, all transmitted DENM messages are delivered without a single message loss. The vehicle is under the Line of Sight of the transmitting RSU, in a short distance, but it is passing under the RSU.

We solved one question: if the speed sensor of a vehicle delivers a wrong measurement, how can it be detected or isolated in a cooperative environment?

As future work, we plan to test our proposal in a real C-ITS environment like the Smart Highway testbed [24].

ACKNOWLEDGMENT

This work has been funded by the European Union Horizon-2020 Project 5G-CARMEN under Grant Agreement 825012. This work was carried out also with the support of the 5G-Blueprint project, funded by the European Commission through the Horizon 2020 programme under agreement No. 952189. The views expressed are those of the authors and do not necessarily represent the project. The Commission is not liable for any use that may be made of any of the information contained therein.

REFERENCES

[1] ETSI EN 302 637-2 V1.4.1. (2019, January) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Messages Basic Service. [Online] Available: https://www.etsi.org/deliver/etsi_EN/302600_302699/30263702/01.04.01_30/en_30263702v010401v.pdf, Last accessed on: May, 2021.

[2] ETSI EN 302 637-3 V1.3.1. (2019, April) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service. [Online] Available: https://www.etsi.org/deliver/etsi_en/302600_302699/30263703/01.03.01_60/en_30263703v010301p.pdf, Last accessed on: May, 2021.

[3] ETSI EN 302 663 V1.3.0. (2019, May) Intelligent Transport Systems (ITS); ITS-G5 Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band. [Online] Available: https://www.etsi.org/deliver/etsi_en/302600_302699/302663/01.03.00_20/en_302663v010300a.pdf, Last accessed on: May, 2021.

[4] ETSI TS 101 539-2 V1.1.1 . (2018, June) Intelligent Transport Systems (ITS); V2X Applications; Part 2: Intersection Collision Risk Warning (ICRW) application requirements specification. [Online] Available: https://www.etsi.org/deliver/etsi_ts/101500_101599/10153902/01.01.01_60/ts_10153902v010101p.pdf, Last accessed on: May, 2021.

[5] ETSI TS 101 539-3 V1.1.1. (2013, November) Intelligent Transport Systems (ITS); V2X Applications; Part 3: Longitudinal Collision Risk Warning (LCRW) application requirements specification. [Online] Available: https://www.etsi.org/deliver/etsi_ts/101500_101599/10153903/01.01.01_60/ts_10153903v010101p.pdf, Last accessed on: May, 2021.

[6] ETSI TR 102 638 V1.1.1 . (2009, June) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions. [Online] Available: https://www.etsi.org/deliver/etsi_tr/102600_102699/102638/01.01.01_60/tr_102638v010101p.pdf, Last accessed on: May, 2021.

[7] J. A. F. F. Dias, J. J. P. C. Rodrigues, F. Xia, and C. X. Mavromoustakis, "A Cooperative Watchdog System to Detect Misbehavior Nodes in Vehicular Delay-Tolerant Networks," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7929–7937, 2015, doi: https://doi.org/10.1109/TIE.2015.2425357.

[8] J. Kamel, A. Kaiser, I. Ben Jemaa, P. Cincilla, and P. Urien, "Feasibility Study of Misbehavior Detection Mechanisms in Cooperative Intelligent Transport Systems (C-ITS)," in *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, 2018, pp. 1–5, doi: https://doi.org/10.1109/VTCSpring.2018.8417867.

[9] ETSI TR 103 415. (2018, April) Intelligent Transport Systems (ITS);Security; Pre-standardization study on pseudonym change management. [Online] Available: https://www.etsi.org/deliver/etsi_tr/103400_103499/103415/01.01.01_60/tr_103415v010101p.pdf, Last accessed on: May, 2021.

[10] Mahmoudi, Issam and Kamel, Joseph and Ben-Jemaa, Ines and Kaiser, Arnaud and Urien, Pascal, "Towards a Reliable Machine Learning-Based Global Misbehavior Detection in C–ITS: Model Evaluation," in *Vehicular Ad-hoc Networks for Smart Cities: Third International Workshop, 2019*, 2020, vol. 1144, p. 73, [Online] Available: https://link.springer.com/chapter/10.1007/978-981-15-3750-9_6, Last accessed on: May, 2021.

[11] United Nations. (2010, May) Regulation No 39 of the Economic Commission for Europe of the United Nations (UN/ECE) — Uniform provisions concerning the approval of vehicles with regard to the speedometer equipment including its installation. [Online] Available: https://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2010:120:0040:0048:EN:PDF, Last accessed on: May, 2021.

[12] *RXS-DR-10 Speed & Range Measurement Radar*, Radarxense BV, 1 2018, ver. 1.2 [Online] Available: https://www.radarxense.com/wp-content/uploads/2018/01/RADARXENSE_Datasheet-RXS-DR-10_V1.2.pdf, Last accessed on: May, 2021.

[13] WEGCODE.BE. Wat is de tolerantiemarge? - What is the tolerance margin. [Online] Available: https://www.wegcode.be/modules/mod_boeteberekening/help/marge_nl.htm, Last accessed on: May, 2021.

[14] P. Große and C. Andrich and W. Kotterman and A. Ihlow and G. D. Galdo, "Measuring ETSI ITS-G5 Communications Latencies with Commercial off-the-shelf Wi-Fi Hardware," in *2018 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM)*, 2018, pp. 1–4, doi: https://doi.org/10.1109/ICMIM.2018.8443355.

[15] Festag, Andreas, "Cooperative Intelligent Transport Systems Standards in Europe," *IEEE Communications Magazine*, vol. 52, no. 12, pp. 166–172, 2014, doi: https://doi.org/10.1109/MCOM.2014.6979970.

[16] Jenoptik. (2020) Monitoring average speed using intelligent average speed cameras. [Online] Available: https://www.jenoptik.com/products/road-safety/average-speed-camera, Last accessed on: May, 2021.

[17] S. Ellwanger and E. Wohlfarth, "Truck platooning application," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 966–971, doi: https://doi.org/10.1109/IVS.2017.7995840.

[18] European Road Safety Observatory. (2018, June) Traffic Safety Basic Facts 2018. [Online] Available: https://ec.europa.eu/transport/road_safety/sites/roadsafety/files/pdf/statistics/dacota/bfs20xx_cyclists.pdf, Last accessed on: May, 2021.

[19] European Transport Safety Council. (2020, January) PIN Flash Report 38. [Online] Available: https://etsc.eu/wp-content/uploads/PIN-Flash-38_FINAL.pdf, Last accessed on: may, 2021.

[20] Raphael Riebl, "VANETZA," https://github.com/riebl/vanetza, 2021.

[21] ETSI EN 302 636-1 V1.2.1. (2014, April) Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 1: Requirements. [Online] Available: https://www.etsi.org/deliver/etsi_en/302600_302699/30263601/01.02.01_60/en_30263601v010201p.pdf, Last accessed on: May, 2020.

[22] ETSI TS 102 687 V1.2.1 (2018-04). (2018, April) Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part. [Online] Available: https://www.etsi.org/deliver/etsi_ts/102600_102699/102687/01.02.01_60/ts_102687v010201p.pdf, Last accessed: May, 2021.

[23] Takayuki Shimizu and Bin Cheng and Hongsheng Lu and John Kenney, "Comparative Analysis of DSRC and LTE-V2X PC5 Mode 4 with SAE Congestion Control," in *2020 IEEE Vehicular Networking Conference (VNC)*. IEEE, December 2020, doi: https://doi.org/10.1109/vnc51378.2020.9318353.

[24] Marquez-Barja, Johann and Lannoo, Bart and Naudts, Dries and Braem, Bart and Maglogiannis, Vasilis and Donato, C and Mercelis, Siegfried and Berkvens, Rafael and Hellinckx, Peter and Weyn, Maarten and others, "Smart Highway: ITS-G5 and C2VX based testbed for vehicular communications in real environments enhanced by edge/-cloud technologies," in *EuCNC2019, the European Conference on Networks and Communications*, June 2019, pp. 1–2, [Online] Available: https://biblio.ugent.be/publication/8642435, Last accessed on: May, 2021.