An Error Correction Algorithm for Forward Collision Warning Applications

João B. Pinto Neto[1,2], Lucas C. Gomes[2], Eduardo M. Castanho[2],
Miguel Elias M. Campista[2], Luís Henrique M. K. Costa[2] and Paulo Cezar M. Ribeiro[3]

***Abstract*— Intelligent Transportation Systems (ITS) are based on the intelligence placed on roadside units and onboard vehicles. ITS technologies, like connected cars, improve road safety by having vehicles communicating with each other, with the infrastructure, or both. The communication uses the 5.9 GHz band, called Dedicated Short Range Communications (DSRC), and protocols defined in the Wireless Access in Vehicular Environments (WAVE) architecture. This paper evaluates, through practical experiments with latest on-board units (OBUs), the performance of a forward collision warning (FCW) application operating in the DSRC control channel number 178. The application uses location information provided by an internal high-precision GPS to calculate the safe braking distance from a vehicle moving towards a stationary vehicle.**

Experiments were conducted at speeds of 30, 40, 50 and 60 km/h with a GPS update rate of 5 Hz. Our results show a margin of error below 1%, demonstrating the required reliability to forward collision avoidance applications.

***Index Terms*— Vehicular networks, WAVE, forward collision warning.**

I. INTRODUCTION

Intelligent Transportation Systems (ITS) have gained significant attention from academia and industry with the emergence of new vehicular communication standards. The WAVE (Wireless Access in Vehicular Environments) architecture establishes a set of patterns that includes the IEEE 1609 family and the 802.11p amendment of the IEEE 802.11 standard [1]. The physical layer uses the frequency band from 5.850 to 5.925 GHz. The so-called Dedicated Short Range Communications (DSRC) is an exclusive band that can be used for communications between vehicles (V2V) and/or between vehicles and fixed infrastructure (V2I). The DSRC band was reserved by the Federal Communications Commission (FCC) of the United States of America and by the European Telecommunications Standards Institute (ETSI). On the other hand, Japan adopted the 700 MHz band [2]. There are many applications intended for the DSRC band, but many of them are active safety applications and other services related to the security of passengers and pedestrians.

Therefore, one of the foundations of ITS is safety. The information exchanged between vehicles through Basic Safety Messages (BSMs) of the WAVE standard, within a defined coverage area, allows a considerable increase of road safety level [3]. The utilization of BSMs allows the implementation of forward collision warning, lane change warning, intersection collision warning, and abrupt braking alarm applications. Those are termed active safety applications because they prevent collisions in a cooperative way, differing from the passive devices, such as airbags and safety belts, that minimize physical and material damages once an accident has occurred. Active safety applications have potential to reduce accidents in expressways [4]. Nevertheless, they require low-latency message exchanges between vehicles. According to Hill [5], the DSRC technology complies with the requirements of active safety applications, because it yields latency in the order of 0.2 microseconds. Thus, the sensing devices of the vehicles must also meet latency demands.

Several works are dedicated to collision warning and emergency braking systems. Tsai et al. [6] propose a cooperative emergency braking warning system which combines DSRC communications with image recognition using a camera installed in the vehicle. The camera images are used to generate a map of the surroundings of the vehicle to identify the message sender, improving positioning accuracy even in bad weather conditions. Chen et al. [7] propose an algorithm for collision avoidance through alert messages to inform the driver of the time-to-collision and safe braking distance based on the speed of the vehicle, acquired from an ultrasound sensor, combined with the driver's reaction time. A high speed real experiment was conducted by Shafiq et al. [4] to develop an adaptive method to register the vehicle's braking time from a known speed until a complete stop to calculate the safe braking distance. Based on the registered value, the safe distance is calculated and updated at every stop event. Other approaches related to driver behavior and use of multisensor techniques to estimate position and speed [8]–[10] are also proposed. Nevertheless, those specifically regarding safety applications diverge from the proposal of the present paper, either because the performance is evaluated by simulations or because they do not use a GPS receiver as a unique distance sensor device to determine the safe braking distance.

In this context, this work evaluates the accuracy of a forward collision warning application by calculating the safe braking distance of a moving vehicle in relation to a stopped vehicle, using latest On-Board Units (OBUs).

[1]Instituto Federal de Educação, Ciência e Tecnologia de Rondônia - IFRO joao.pinto@ifro.edu.br
[2]Universidade Federal do Rio de Janeiro - PEE/COPPE/GTA {gomes,castanho,miguel,luish}@gta.ufrj.br
[3]Universidade Federal do Rio de Janeiro - PET/COPPE pribeiro@pet.coppe.ufrj.br

The application uses the positions provided by the GPS embedded in the OBUs to emit an audio or video signal to the driver, considering the geographic position, the direction of movement, and the vehicle speed. A challenge is produced by the use of a GPS as the source of the distance and vehicle speed information to calculate the safe braking distance. The embedded GPS update period (t_{GPS}) is 200 ms, which is twice the latency required for a forward collision warning application, and could as a consequence compromise its precision. The analysis of the data obtained in initial experiments showed that t_{GPS} is the main cause of warning braking distance calculation errors. Therefore, to minimize the influence of t_{GPS} on the application accuracy, an algorithm to correct the forward collision warning errors, named ACORE (Algorithm for COllision waRning Error correction), is developed in this paper. The algorithm computes the time required to reach the safe braking distance as a function of the vehicle's speed. The results obtained show a margin of error below 1%, demonstrating that the required reliability to forward collision avoidance applications can be guaranteed by using ACORE.

This work is organized as follows. Section II describes the operation of the forward collision warning application. Section III presents in detail the proposed algorithm to correct the GPS update rate. Next, Section IV describes where and how the experiments were carried out. Then, in Section V, the experimental results are presented and finally, Section VI concludes the paper with some remarks and presents future work.

II. Application for the Safe Braking Distance Calculation

The goal of the application is to alert the driver of the vehicle of the potential occurrence of a forward collision with a stationary vehicle in the same lane. The application running in the stationary vehicle obtain the current position of the vehicle, its current speed, elevation, heading, time from the embedded GPS and vehicle characteristics, and encoding this data to construct a BSM, using the ASN.1 standard [11].


```
<BasicSafetyMessage>
  <msgID><basicSafetyMessage/></msgID>
  <blob1>
    74 2E 7C 0E 21 AB E0 F2 5E FC 6C E6 3C 8F 5C 00
    63 FF FF FF FF 2B D0 2D 7F 05 00 8C 00 1E 14 00
    AF 00 00 32 C1 A4
  </blob1>
  <safetyExt>
    <events>256</events>
  </safetyExt>
</BasicSafetyMessage>
```


Fig. 1: Example of a typical Basic Safety Message (BSM).

Figure 1 shows the content of the BSM message sent over the control channel by the stationary vehicle. The mobile vehicle application receives and decodes the BSM and using the information of its own embedded GPS calculates the safe brake distance, monitors position and vehicle speed.

The character string (`blob1`) contains, among other encoded information, a message identifier (`0x74`/116), a time stamp (`0xAB E0`/4400 ms), the car's latitude (`0xF2 5E FC 6C`/-22.8656020 degrees), longitude (`0xE6 3C 8F 5C`/-43.2238756 degrees), elevation (`0x00 63`/9.9 m), speed (`0x2B D0`/11.216 m/s), heading (`0x2D 7F`/145.8824 degrees), and dimensions (width/length) (`0x32 C1 A4`/812 cm/420 cm), highlighted on Figure 1. These messages are sent over the DSRC Control Channel (178) every 50 ms to the vehicles inside of the radio range.

The applications used in this work was developed in C language, using the Software Development Kit (SDK) provided by Cohda Wireless [12], the OBU manufacturer. When a new message is received, the application calculates the distance between the units and compares it with the safe braking distance, whose main component is given by $D_b(v)$, the distance required to completely stop the vehicle moving at speed v [7]. The distance $D_b(v)$ is computed as follows:

$$D_b(v) = \frac{\gamma W}{2gC_{ae}} ln(1 + \frac{C_{ae}v^2}{\eta(\mu + f_r)W cos\theta + W sin\theta}), \quad (1)$$

where $C_{ae} = (\rho A_f C_d)/2$. The remaining parameters and their respective values are given in Table I.

TABLE I: Parameters and values used to calculate D_b.

Parameter	Description	Values
γ	Equivalent mass factor	1.04 kgm^2
g	Acceleration of gravity	9.80 m/s^2
ρ	Mass density of the air	1.30 kg/m^3
A_f	Characteristic area of the vehicle	2.24 m^2
C_d	Coefficient of aerodynamic resistance	0.35
η	Brake efficiency	0.9
μ	Road adhesion coefficient	0.75
f_r	Rolling resistance coefficient	1.04
W	Vehicle Weight	1050 Kg
θ	Angle of the road slope with the horizontal	0°

It is also necessary to take into account the distance traveled during the driver's reaction time (D_r) and the distance covered during the time of the effective acting of the braking system (D_p). This last distance is accounted after the foot pedal is depressed by the driver [7]. These times vary within the intervals of 0.74 to 1.7 s and 0.3 to 0.7 s, respectively. Therefore, the safe braking distance (D_{safe}) is the sum of three terms:

$$D_{safe} = D_b + D_r + D_p. \quad (2)$$

The values used in the application are 1.0 s to the driver's reaction time and 0.5 s to the effective brake acting. It is important to note that using Equations 1 and 2 with exactness is not a concern, since that is not the focus of this study. The estimated calculation of D_{safe} does not compromise the validity of the experiments, that will yield equivalent results when executing the calculations with exact parameter values. Thus, parameters like γ, η, ρ, μ, and f_r are configured as the average value of their respective variations. Nevertheless, in the final implementation of an active safety application, other factors like the use of Anti-lock Braking System (ABS) or the weather conditions that can alter the coefficient of friction between track and tires, visibility and the horizontal precision of the GPS, related to latitude, longitude, and speed, must be taken into account.

The mobile unit updates the safe braking distance as a function of the speed, latitude, and longitude values provided by the internal GPS. Using this information combined with the data received from the BSM sent by the stationary unit, the mobile unit calculates the current distance (D_{act}) between the two units based on the Haversine formula [13]:

$$D_{act} = 2R * \arctan(\sqrt{a/1-a}), \tag{3}$$

where R is the Earth's radius in meters, $a = (\sin^2(\Delta Lat/2) + \cos(LatA * C) * cos(LatB * C) * sin^2(\Delta Lon/2)$, $C = \pi/180$, $\Delta Lat = (LatA - LatB) * C$, and $\Delta Lon = (LonA - LonB) * C$.

The application compares, each time a BSM is received, the distance D_{act} with the distance D_{safe}. If the distance is equal or smaller than the safe distance ($D_{act} \leq D_{safe}$), an alert is emitted and the driver must start braking. Our field experiments confirmed that the GPS update rate is the cause of collision alert errors. Considering the GPS update period of 200 ms adopted in our tests, it was possible to verify that alerts occur after the vehicle travels distances varying from 1.1 to 8.3 meters, corresponding to speeds from 20 to 150 km/h, respectively. These distances can be greater in case of an interruption in the sequence of the received messages, due to packet losses for example. The 200 ms update period was used because that is the minimum period supported by the embedded GNSS receiver, Ublox NEO-M8N, for both GPS and GLONASS reception modes.

III. Algorithm for Collision Warning Error Correction (ACORE)

To minimize the influence of the GPS update rate, the Algorithm for Collision waRning Error correction (ACORE) is proposed. It anticipates the warning of safe braking distance in function of the GPS update rate. To illustrate the operation of ACORE, a part of the round 13 of the 40 km/h session (of the experiments described later) was selected. In Figure 2, the X-axis shows the elapsed time, within a 900 ms window, whereas the Y-axis shows the distance in meters. The ladder-shaped blue curve represents the distance between the mobile and the stationary unit (D_{act}), and the blue dots correspond to the instants in which the BSMs are received. The dashed red curve represents the safe braking distance while the red dots stand for the instants in which the BSMs are received, as well. Points A, B, C, and D represent the instants when coordinates and speed information are updated by the GPS.

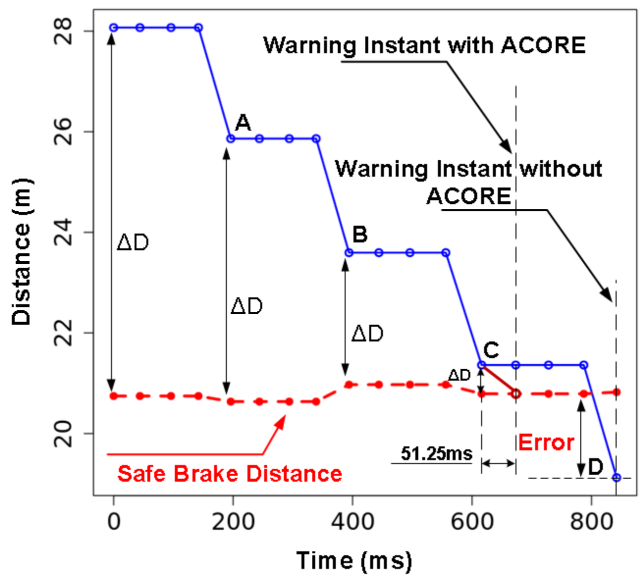


Fig. 2: Error Correction executed by the Algorithm for Collision Warning Error Correction (ACORE).

Each time the mobile unit receives a BSM, the application calculates the distance between the units (D_{act}) using Equation 3, and the safe braking distance (D_{safe}) using Equation 2. The former is a function of the geographic coordinates, while the latter is a function of the speed, both provided by the GPS. One can observe in the time lapse between points A and B the reception of 4 messages with the same coordinates, despite the mobile continues in its movement. This "blind" interval is repeated from points B, C, and finally in point D, after the mobile unit overcomes the safe braking distance, producing an error of about 1.70 m, shown in Figure 2.

Algorithm 1: ACORE algorithm.

Input: GPS update period (t_{GPS}), current distance (D_{act}), safe braking distance (D_{safe}), current speed (v)
Output: time to Warning (t_W)

begin
  $\Delta D \leftarrow D_{act} - D_{safe}$
  $t_W \leftarrow \Delta D / v$
  if $t_W \leq t_{GPS}$ **then**
    return(t_W)
  end
end

According to Algorithm 1, at every incoming BSM, ACORE checks the difference between the current position and the safe braking distance (ΔD) and calculates the time interval to reach the safe braking distance considering the current speed. If the obtained time to warning (t_W) is smaller than the GPS update period (t_{GPS}), the application triggers a collision warning after t_W seconds. The point C in Figure 2 corresponds to the time instant 616 ms when t_W (51.25 ms) is less than t_{GPS} (200 ms). The alert is triggered at the instant 667 ms, correcting the error that would happen without applying ACORE. Without ACORE, the alert would be triggered only at instant 846 ms (point D).

It is important to observe that the accuracy of ACORE depends on a constant speed during the time to warning t_W. The algorithm assumes a constant speed during this time interval to guarantee a null error. Considering an acceleration of 0 to 100 km/h in 3.6 s during a time interval of 200 ms, the error obtained with ACORE would be of 15 cm. That error would be negligible, even considering this worst-case acceleration that is only reached by sport cars. The accuracy of ACORE relies on the GPS signal to estimate location and speed. If the GPS signal is not reliable, the Forward Collision Warning application must operate using other alternatives. Data acquisition can be obtained, for instance, from a Controller Area Network (CAN) based interface [14] already available at the vehicle.

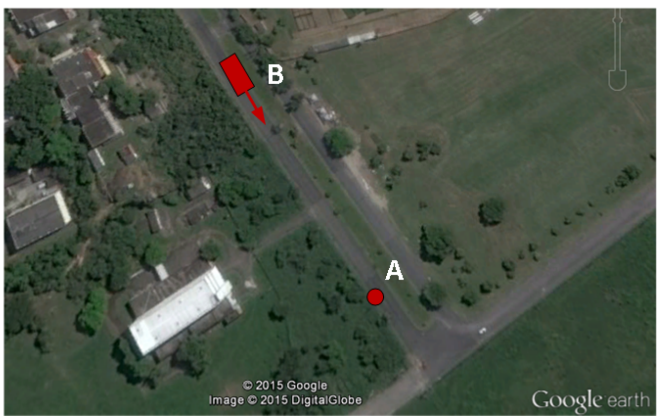


Fig. 3: Overview of the site of experiments (source: Google Earth).

IV. Experimental Methodology

The performance of the application is evaluated based on experiments conducted at the campus of Federal University of Rio de Janeiro (UFRJ). The scenario used within UFRJ is illustrated in Figure 3. It consists of a stationary unit and a mobile unit moving from point B to point A, equipped with the IEEE 802.11p communication devices.

TABLE II: Hardware description.

Hardware	Description
OBU	Cohda Wireless model MK5
DSRC Antenna	2 x 5,9 GHz Omnidirecional MobileMark ECO6-5500e
GPS Antenna	1 x WELL-HOPE GPS/GLON-09B

A. Experimental Setup

The hardware description is provided in Table II, whereas the OBU settings are listed in table III. The point A of Figure 3 corresponds to the stationary unit mounted on the roadway edge, equipped with a Cohda MK5 OBU. In this unit, antennas were installed at a height of 1.40 m. An application was configured to continuously send BSMs every 50 ms over the control channel. Point B, on the other hand, corresponds to the mobile unit, mounted in a passenger car, whose aerodynamic and dimensional features are included as parameter settings of the application. In this unit, antennas were mounted on the car roof (Figure 4). The experiments were performed on Saturday mornings when the university campus is almost empty, i.e., with low traffic flow, clear sky, temperature ranging between 23 and 29^{o}C, and relative humidity ranging between 66 and 88%.

TABLE III: OBU settings.

Parameter	Configuration
DSRC channel	178
Transmission power	20 dBm
Data rate	6 Mbps
Message length	51 Bytes
BSM's sending interval	50 ms
GPS update period	200 ms

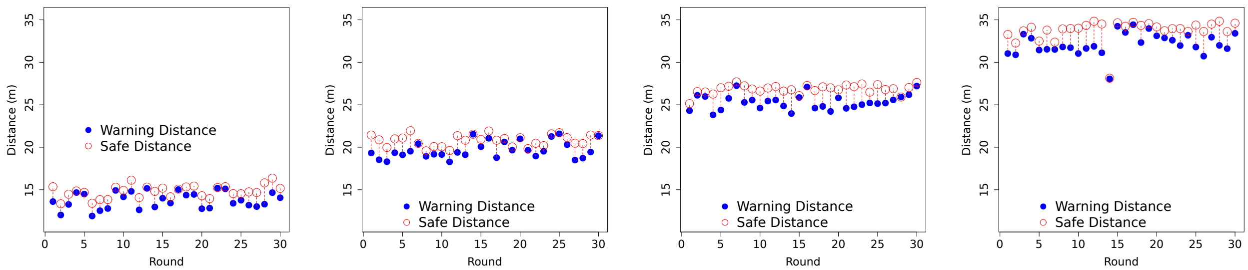

Fig. 4: Antenna module.

B. Experimental Procedure

Four experimental sessions were conducted, corresponding to different average speeds of 30, 40, 50, and 60 km/h. In each session, 30 rounds were conducted and all the BSMs sent by the stationary unit were stored. The SAE J2735 standard [15] states that for an active safety application, the control channel must be monitored, at least, every 100 ms. This period must be smaller in traffic congestion to avoid packet losses that could completely lose track of neighbor vehicles [16]. Thus, In our experiments, we forced the OBUs to send and receive messages every 50 ms in order to verify channel realiability.

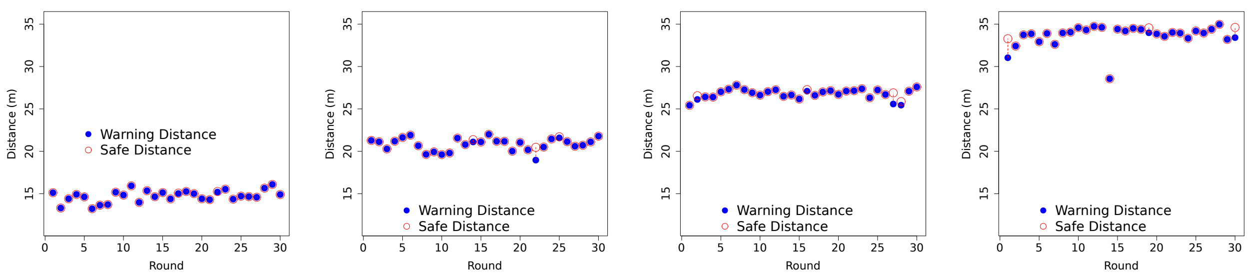
V. Experimental Results

Figures 5 and 6 show the performance of ACORE along time. These figures present the safe braking distance and the



(a) 30 km/h. (b) 40 km/h. (c) 50 km/h. (d) 60 km/h.

Fig. 5: Safe Distance versus Warning Distance without ACORE.



(a) 30 km/h. (b) 40 km/h. (c) 50 km/h. (d) 60 km/h.

Fig. 6: Safe Distance versus Warning Distance with ACORE.

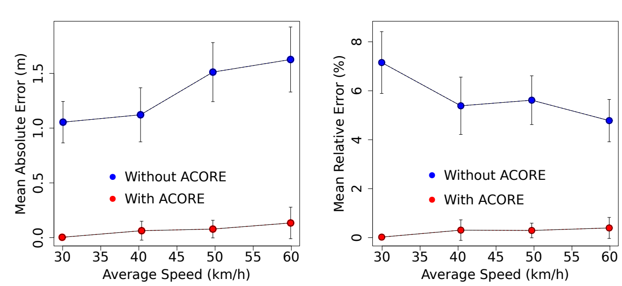
warning braking distance obtained from the 30 rounds of each speed session.

In all the plots, the Y-axis represents the distance, in meters, whereas the X-axis represents the round number. The empty red circles represent the safe distance and the full blue circles represent the warning distance. The length of red dashed lines represents the error, in meters, of the safe braking warning distance.

Figure 5 shows the comparison between the warning distance calculated by the application and the braking distance, without applying ACORE. Significant errors for all speeds, with no observable regular pattern, are obtained. Nonetheless, we can note that there is greater incidence of such errors at higher speeds (50 and 60 km/h). Figure 6 shows that these errors are suppressed by ACORE: almost completely at 30 km/h, where we have the best session in terms of ACORE's performance. In the 40 km/h session, ACORE also shows good performance, with the exception of round 22, at which a more relevant error occurs. Although the performance slightly deteriorates at higher speeds, 50 and 60 km/h, ACORE still shows very closer distances as compared with the results of Figure 5 (without ACORE).

A. ACORE Error Analysis

The analysis of the mean absolute error as a function of speed (Figure 7a), for the application without ACORE, reveals a growing trend. This is expected, due to the relationship of direct proportionality between the braking distance and speed, reaching a value of 1.6 m at 60 km/h. Applying ACORE, the average absolute error was as low as 15 cm, confirming the efficiency of the algorithm, regardless of speed. Figure 7b shows the relationship between average relative error and vehicle speed. One can observe the inverse relationship in comparison with the absolute error without ACORE. This downward trend is justified because the relative error and the braking distance are inversely proportional. The results obtained by ACORE reduced the mean relative error rate to below 1% in all speeds. The figure also shows that a greater mean relative error corresponds to the occurrence of greater dispersion of values from the mean value, which happens at 40 and 60 km/h, due to errors not corrected by ACORE.



(a) Mean Absolute Error. (b) Mean Relative Error.

Fig. 7: Safe Distance Error Versus Average Speed.

The significant maximum average error drop forced by ACORE in the 30 km/h session was caused by the correction of all relevant errors, proving the efficiency of the algorithm in this session. It is important to remark that there were no messages losses or messages received out of sequence in any session.

VI. Conclusion and Future Works

In this paper, we proposed ACORE (Algorithm for COllision waRning Error correction), an error correction algorithm designed to improve the performance of Forward Collision Warning applications using the WAVE architecture and compatible equipment. Field experimental results showed that, without ACORE, the efficiency and reliability of the application would be compromised. Moreover, the obtained results confirmed ACORE's capability to correct errors introduced by typical GPS receivers operating as sensing devices for active safety applications. The use of a GPS as safe braking distance measurement device brings the benefit of avoiding the deployment of sensing equipment in addition to the OBUs. The average relative error margin drops below 1% using ACORE, ensuring reliability to Forward Collision Warning applications using the WAVE/DSRC technology.

As future work, we plan to propose new versions of ACORE taking into account acceleration within the GPS update period and increase the application's priority to avoid convergence errors. A model should be developed to assess the Forward Collision Warning Application accuracy related to latency and message loss under high vehicle density scenarios. In addition, the model will enable evaluating the performance of ACORE at higher speeds. Another goal is to carry out experiments under adverse weather conditions, absence of line of sight, and GPS signal loss in scenarios such as dense forest areas and tunnels.

Acknowledgments

This work was partially funded by IMI project: Intelligent Mobile Infrastructure based on Cellular and Vehicular Networks - FAPERJ, by CNPq, CAPES, and FAPERO.

References

[1] "IEEE Guide for Wireless Access in Vehicular Environments (WAVE) - Architecture," *IEEE Std 1609.0-2013*, pp. 1–78, March 2014.

[2] L. Delgrossi and T. Zhang, *Dedicated Short-Range Communications*. John Wiley & Sons, Inc., 2012, pp. 44–51.

[3] W. Liu and B. Li, *Research on the Mechanism of Intelligent Transportation Systems on Improving Road Safety*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 257–263.

[4] Z. Shafiq, L. U. Khan, S. A. Mahmud, and G. M. Khan, "Context-aware cooperative collision avoidance vehicle braking alert system for VANET," in *Proceedings of the 3rd International Conference on Context-Aware Systems and Applications*, ser. ICCASA '14, ICST, Brussels, Belgium, pp. 92-96, 2014.

[5] C. Hill, "ITS ePrimer Module 13: Connected Vehicles," 2015, united States Department of Transportation. [Online]. Available: https://www.pcb.its.dot.gov/ [Accessed: 2015-12-05]

[6] 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, no. 1, pp. 1–14, 2015. [Online]. Available: http://dx.doi.org/10.1186/s13638-015-0262-0 [Accessed: 2105-12-10]

[7] Y.-L. Chen, K.-Y. Shen, and S.-C. Wang, "Forward collision warning system considering both time-to-collision and safety braking distance," in *Industrial Electronics and Applications (ICIEA), 2013 8th IEEE Conference on*, ser. pp. 972-977, June 2013.

[8] X. Song, X. Li, and W. Zhang, "Key parameters estimation and adaptive warning strategy for rear-end collision of vehicle," *Mathematical Problems in Engineering*, vol. 2015, 2015.

[9] J. Wang, C. Yu, S. E. Li, and L. Wang, "A forward collision warning algorithm with adaptation to driver behaviors," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1157–1167, April 2016.

[10] N. Alsbou, N. Martinek, B. Beckrest, and N. Pax, "Vehicle collision avoidance system," in *Proceedings of the 2015 ASEE North Central Section Conference*, 2015.

[11] J. Missener and D. Kelley. (2015) ASN.1 Specification – Official normative ASN.1 of the SAE J2735. SAE International. [Online]. Available: http://www.sae.org/standardsdev/dsrc/usa/ [Accessed: 2015-10-15]

[12] COHDA, "Cohda Mobility MKx SDK," 2015, COHDA Wireless Pty, ltd. [Online]. Available: http://cohdawireless.com/ [Accessed: 2015-08-27]

[13] F. Ivis, "Calculating geographic distance: Concepts and method," 2006. [Online]. Available: http://www.lexjansen.com/nesug/nesug06/dm/ [Accessed: 2015-12-05]

[14] G. Sood, J. Willis, A. Jaekel, S. Nazeer, and D. Paulraj, "Analyzing accuracy of gps data for vehicular parameters," in *2015 International Conference on Connected Vehicles and Expo (ICCVE)*, Oct 2015, pp. 344–345.

[15] D. Kelley. (2015) DSRC implementation guide. SAE International. [Online]. Available: http://www.sae.org/standardsdev/dsrc/ [Accessed: 2015-10-25]

[16] L. Shi and K. W. Sung, "Spectrum requirement for vehicle-to-vehicle communication for traffic safety," in *2014 IEEE 79th Vehicular Technology Conference (VTC Spring)*, May 2014, pp. 1–5.