Paper number AM-SP0222

An Algorithm for Error Correction in Forward Collision Warning Application

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

1. Universidade Federal do Rio de Janeiro - PEE/COPPE/GTA, Brazil

2. Instituto Federal de Educação, Ciência de Tecnologia de Rondônia – IFRO, Brazil

3. Universidade Federal do Rio de Janeiro - PET/COPPE, Brazil

*. Rio de Janeiro, 22775-036, Brazil, +55 21 99730-0804, pinto@gta.ufrj.br

Abstract

Connected vehicles are a key technology to Intelligent Transportation Systems (ITS), with applications between vehicles, and between vehicles and infrastructure and both two techniques. 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 research evaluates, through practical experiments with latest on-board units (OBUs), the performance of a forward collision warning (FCW) application sending and receiving messages over the DSRC control channel 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. The proposed algorithm limits the error within the GPS update rate. The experiments were conducted at speeds of 20, 30, 40, 50 and 60 an hour with GPS update rate of 5 Hz. The results show a margin of error below 1%, proving the reliability of the application for avoiding forward collisions.

Keywords:

Vehicular communications, DSRC, forward-collision warning.

Introduction

Intelligent Transportation Systems (ITS) have gained significant attention from the academia with the emergence of new vehicular communication standards. The WAVE (Wireless Access in Vehicular Environments) architecture establishes a set of standards that include the IEEE 1609 family and the amendment 802.11p of the IEEE 802.11 standard [1]. The physical layer uses the frequency band of 5,850 to 5,925 GHz. This exclusive band can be used for communications between vehicles (V2V) and/or between vehicles and fixed infrastructure (V2I), being called Dedicated Short Range Communications (DSRC). This 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 intended applications for the DSRC band, but active security applications receive huge attention.

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 to the drivers' security. The use of BSMs allows the implementation of forward collision warning, lane change warning, intersection collision warning and abrupt braking alarm applications. Those are called active safety applications because they prevent collisions in a cooperative way, differently from passive devices, like airbags and safety belts that minimize physical and material damages. Active safety applications have potential to reduce accidents in expressways [3]. Nevertheless, they demand a low latency in the message exchanged between vehicles. In most cases, the extremely low delays required by traffic safety applications demonstrate the need for ad-hoc network architectures with support to direct vehicle-to-vehicle communication [4]. According to Hill **[**5**]**, the DSRC technology complies with the requirements of active safety applications, because it yields latencies in the order of 0.2 microseconds. Thus, the sensing devices of the vehicles must also meet the latency demands.

In this context, this work evaluates the accuracy of a forward collision warning application calculating the safe braking distance of a moving vehicle in relation to a stopped vehicle, using latest commercial On-Board Units (OBUs). The application uses positions provided by the GPS embedded in the OBUs to emit an audio or video signal to the driver based on the geographic position, on the direction of the movement and on the vehicle speed.

An important challenge is the use of the GPS as the source of the distance and vehicle velocity information to calculate the safe braking distance. The embedded GPS update rate ($\boldsymbol{t_{GPS}}$) is 200 ms. That is twice the latency requirement for a forward collision warning application, and therefore could compromise its precision. The analysis of the data obtained in initial experiments showed that $\boldsymbol{t_{GPS}}$ was the main cause of the errors of the warning braking distance calculation. To minimize the influence of $\boldsymbol{t_{GPS}}$ on the application accuracy, an algorithm to correct the forward collision warning error, named ACORE, was developed. The proposed algorithm computes the required time to reach the safe braking distance in function of $\boldsymbol{t_{GPS}}$. The results obtained shown the margin of error below 1%, proving the reliability of the forward collision avoidance application.

Related Work

The papers presented in this section belong to two categories: the first on collision avoidance alert systems and the second about works which target latency requirements and message refresh rate.

Chen et al. [6] propose an algorithm to take into account the time to collision and safe braking distance. The work describes the development of frontal collision avoidance alert systems and precisely simulates the scenario of a vehicle moving in a highway, running at speeds between zero and 120 km/h. The results show the efficiency of an algorithm for collision avoidance through message alerts of safe braking distance.

Shafiq et al. [3] propose a method to determine the necessary safe distance between vehicles in freeways as a function of their speed. The method uses an adaptive algorithm to register the vehicle’s braking time from a known speed until complete stop. Based on the registered value the safe distance is calculated and changed at every stop event. The field tests were taken in the Peshawar-Islamabad highway, in Pakistan, with speeds between 50 and 170 km/h. The results show great utility to collision avoidance system, especially in cases of the conductor’s fatigue.

Zinchenko et al. [7] evaluate the performance of V2V applications using the DSRC frequency bands based on MATLAB simulations, received power and packet deliver rate data, obtained from real experiments resultant of the communication between a mobile unit and a stationary unit in intersections. The work is focused on the reliability of the V2V safety applications, i.e., how accurate the application must be to not compromise the integrity of the vehicle’s conductor. The authors conclude that the traffic conditions, the presence of vegetation along the roads and the intersection’s topology must be taken into consideration on V2V applications project. The simulations demonstrate that factors as line of sight, vegetation and traffic compromise the reliability of the applications by increasing the message refresh rate.

All of the above presented related work was relevant to the present research. However, those regarding specifically safety applications diverge from the proposal of this study because the performance is evaluate by simulations or because of the use of GPS to determine the safety braking distance. This research is based on an experimental evaluation with commercial WAVE equipment in order to complement the state-of-the-art.

Application for the Safe Braking Distance Calculation

The application’s goal is to alert the driver about the potential occurrence of a forward collision with a stationary vehicle in the same lane. The process consists of obtaining the actual position, the speed and the time from the embedded GPS, and encoding this data with other vehicle information to construct a BSM, using the ASN.1 standard [8]. These messages are sent over the DSRC Control Channel (178), every 50 ms, to the vehicles inside the radio range. The application used in this work was developed in the C language, using the Software Development Kit (SDK) supplied by Cohda Wireless [9], the company that produced the equipment used in the experiments.

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 $\boldsymbol{D_b}(v)$ **(**Eq. 1**)**, the distance required to completely stop the vehicle moving at speed $\boldsymbol{v}$ [10]. This distance can be computed as:

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

where $\boldsymbol{C_{ae} = \rho A_f C_d}$.

Table 2 shows the remaining parameters and the values that were used to compute $\boldsymbol{D_b}(v)$.

Table 2. Active Safety Latency Requirements (Source: [6])

Parameter	Description	Used Value
γ	Equivalent mass factor	1.04 kgm^2
g	Acceleration of gravity	9.8 m/s^2
ρ	Mass density of the air	1.3 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 degree

Moreover, it is necessary to take into account the distance travelled during conductor's reaction time ($\boldsymbol{D_r}$) and the distance covered during the time of the effective acting of the braking system ($\boldsymbol{D_p}$), after the foot pedal is released by the driver **[**10**]**. 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 is the sum of the three distances:

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

The values used in the application are 1.0 s to the conductor's reaction time and 0.5 s to the effective braking acting. It is important to note that using Equations 1 and 2 with exactness was not a concern, as it was not the focus of this study. The estimated calculation of $\boldsymbol{D_{safe}}$ 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, ρ, η, μ and f_r, are configured as the average value of their respective variations. Nevertheless, in the actual implementation of an active safety application, other factors like the use of Anti-lock Braking System (ABS), weather conditions that can alter the coefficient of friction between track and tires, visibility and the horizontal precision of the GPS, related to latitude and longitude must be considered.

The mobile unit updates the safe braking distance in 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 actual distance ($\boldsymbol{D_{act}}$) between the two units using Equation 3, based on the *haversine* formula **[**10**]**.

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

where R is the Earth's radius in meters, a = (sen²(ΔLat/2) + cos(LatA*C)cos(LatB*C)sen²(Δ Lon/2)), C = π/180, ΔLat = (LatA - LatB)*C and ΔLon= (LonA - LonB)*C .

The application compares, each time a BSM is received, the distance $\boldsymbol{D_{act}}$ with the distance $\boldsymbol{D_{safe}}$. If the value is equal or smaller, an alert is emitted and the conductor must start braking. In the experiments it was confirmed that the GPS update rate is the cause of the errors in the collision alert errors. Considering the GPS update rate of 200 ms adopted in the tests, it was possible to verify that the alert occurs after the vehicle travels distances varying from 1,1 to 8,3 meters, corresponding to speeds from 20 to 150 km/h. These distances can be greater in case of a break in the sequence of the received messages due to the packet loss.

Algorithm for Collision Warning Error Correction (ACORE)

To minimize the influence of the GPS update rate, an Algorithm for COllision waRning Error correction (ACORE) was 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 was selected. In the graph of Figure 4 the horizontal axis shows the elapsed time, normalized within a 900 ms window and the vertical axis shows the distance in meters. The ladder-shaped blue curve represents the distance between the mobile and the stationary unit (**$\boldsymbol{D_{act}}$**), and the blue dots correspond to the instants in which the BSMs were received. The dashed red curve represents the safe braking distance, and the red dots stand for the instants in which the BSMs were received, as well. The A, B, C and D points represent the instants when coordinates and speed are updated by the GPS. Each time the mobile unit receives a BSM, the application calculates the distance between the units ($\boldsymbol{D_{act}}$) using Equation **3** and calculates the safety braking distance ($\boldsymbol{D_{safe}}$), using Equation 2. The first is function of the geographic coordinates, and the former is function of the speed, both provided by the GPS. One can observe in the time gap between points A and B the reception of 4 messages with the same coordinates, despite the mobile continues in its movement.

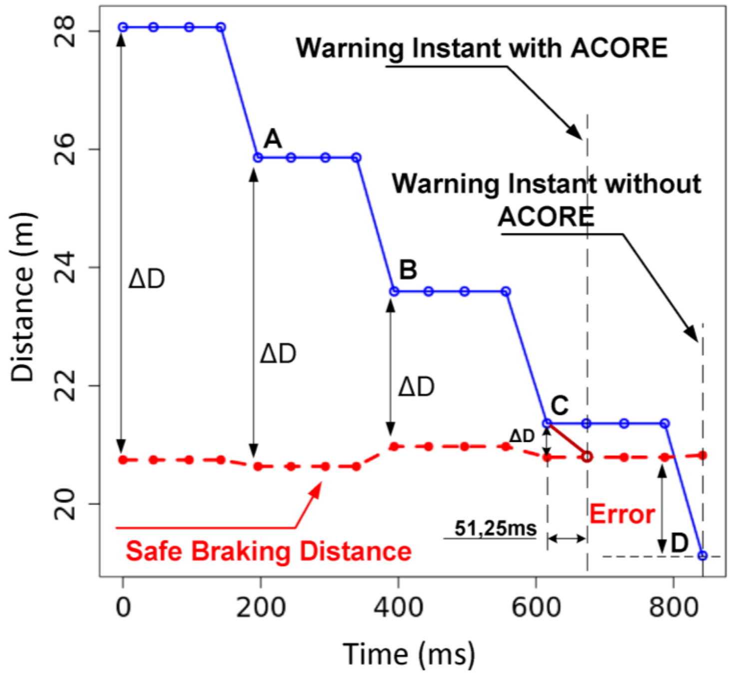


Figure 4. Error-correcting process executed by ACORE.

This "blind" interval is repeated from the points B, C and finally in point D, after the mobile unit overcomes the safe braking distance, producing an error about 1,70 m, shown in Figure 4.

Algorithm 1: Algorithm for Collision Warning Error Correction - ACORE.

Input: GPS update rate (t_{GPS}), actual distance between units (D_{act}), Computed safe distance (D_{safe}), actual speed (v)
Output: Warning timer
begin
  $\Delta D = D_{act} - D_{safe}$
  $\Delta t = \Delta D / v$
  if $\Delta t \leq t_{GPS}$ **then**
    Schedule collision warning to Δt *s*
  end
end

Algorithm 1 describes the Algorithm for Collision Warning Error Correction – ACORE. At every incoming BSM, ACORE checks the difference between the actual position and the safe braking distance ($\boldsymbol{\Delta D}$) and calculates the time interval to reach the safe braking distance regarding the actual speed. If the obtained time interval ($\boldsymbol{\Delta t}$) is smaller than the GPS update rate ($\boldsymbol{t_{GPS}}$), the application triggers a collision warning after $\boldsymbol{\Delta t}$ s. Point C corresponds to the normalized time instant 616 ms in which $\boldsymbol{\Delta t}$ (51,25 ms) is less than $\boldsymbol{t_{GPS}}$ (200 ms). The alert is triggered at instant 667 ms, correcting the error that, without applying ACORE, the alert would trigger only at instant 846 ms (Point D).

It is important to observe that the accuracy of ACORE depends of speed's invariability during the period of time $\boldsymbol{\Delta t}$. 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 centimeters. That error would be negligible, even considering this acceleration that is only reached by supercars.

Experimental Methodology

The application's performance evaluation was based on field experiments conducted at Federal University of Rio de Janeiro (UFRJ), as shown in Figure 5. The evaluation application's scenario consisted of a stationary unit and a mobile unit moving from point B to point A.

Figure 5. Site of the experiments - UFRJ (Source: Google Earth).

The point A corresponds to the stationary unit mounted on the roadway edge, equipped with an OBU. In this unit, antennas were installed at a height of 1.40 m and an application was configured to continuously sending BSMs each 50 ms over DSRC channel 178.

The point B corresponds to the mobile unit, mounted in a passenger car, which aerodynamic and dimensional features are included in the application's parameters configuration. In this unit, the antennas were installed on the car ceiling.

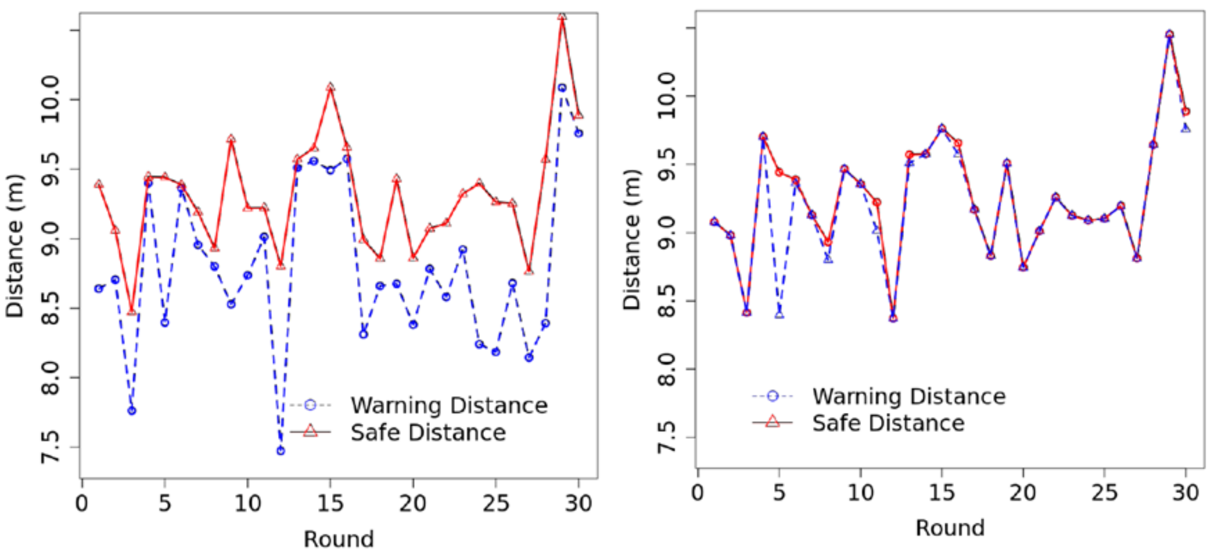
The experiments were performed on Saturdays, with low traffic flow, when the university campus is almost empty, in the morning with clear sky, temperature ranging between 23 and 29 ^{0}C and relative humidity ranging between 66 and 88%.

Experimental Procedure

Five experimental sessions were conducted corresponding to the average speeds of 20, 30, 40, 50 and 60 km/h. In each session were made 30 repetitions (rounds), storing BSMs sent by the stationary unit. The SAE J2735 standard [11] specifies for an active safety application, the control channel must be monitored every 100 ms for a minimum required amount of time. This rate must be smaller in traffic congestion to avoid packet loss that could completely lose track of neighbor vehicles [12]. In the experiments, the OBUs sent messages every 50 ms.

Experimental Results

To illustrate the evolution of the experiment and the performance of ACORE, the sessions of 20 km/h and 40 km/h were selected. The corresponding graphs show the safe braking distance and the warning braking distance superimposed for 30 rounds of each session. The selection of the sessions did not attend any particular criteria, and other sessions showed similar behavior. In the graphs relate to average values, 95 % confidence intervals were used , corresponding to the Student's t-distribution for 30 samples shown by vertical bars in the graphs.



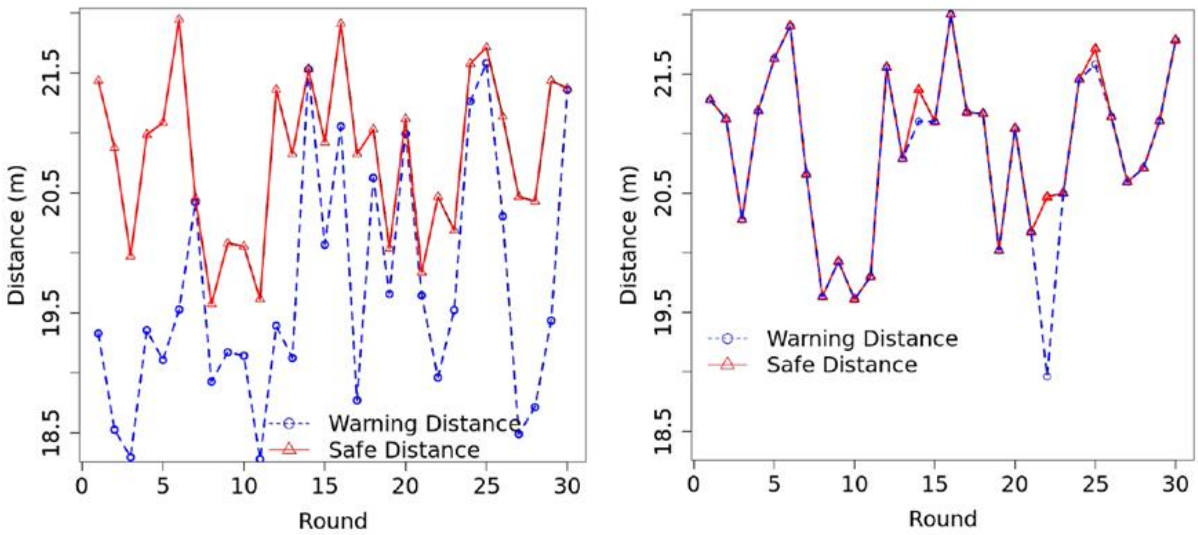
a) Without ACORE **b) With ACORE**

Figure 7. Safe Braking Distance versus Warning Braking Distance at 20 km/h.

Figure 7 shows a comparison between the safe braking distance computed by the application and the measured warning braking distance. Significant errors can be observed in the 4 and 12 rounds (Figure 7a). These errors are suppressed by ACORE, in the same way as other errors between rounds 16 and 30, as can be seen in Figure 7b. However, at round 5, ACORE failed to correct the error because, in this case, $\boldsymbol{\Delta t}$ is greater than $\boldsymbol{t_{GPS}}$, i.e., on calculating $\boldsymbol{\Delta t}$ the resulting value is greater than 200 ms. Thus, in the next update, the mobile unit had already exceeded the safe braking distance, yielding the error shown in the graph.

The 40 km/h session showed a greater difference between the distances than the 20 km/h session, with significant errors at rounds 3, 11, 13, 17 and 27 (Figure 8a). Similarly to the results shown in the previous session (20 km/h), ACORE suppressed all relevant errors, except for round 22, where the value of $\boldsymbol{\Delta t}$ also exceeded the value of the GPS refresh rate (Figure 8b).

The rounds where ACORE failed to correct the errors that occurred for values of $\boldsymbol{\Delta t}$ near the GPS refresh rate like $\boldsymbol{\Delta t}$ = 258 ms and $\boldsymbol{\Delta t}$ = 230 ms, corresponding to rounds 5 and 22 of the 20 and 40 km/h sessions, respectively. These and other similar incidents, found in other sessions, were responsible for the errors shown in Figure 9.

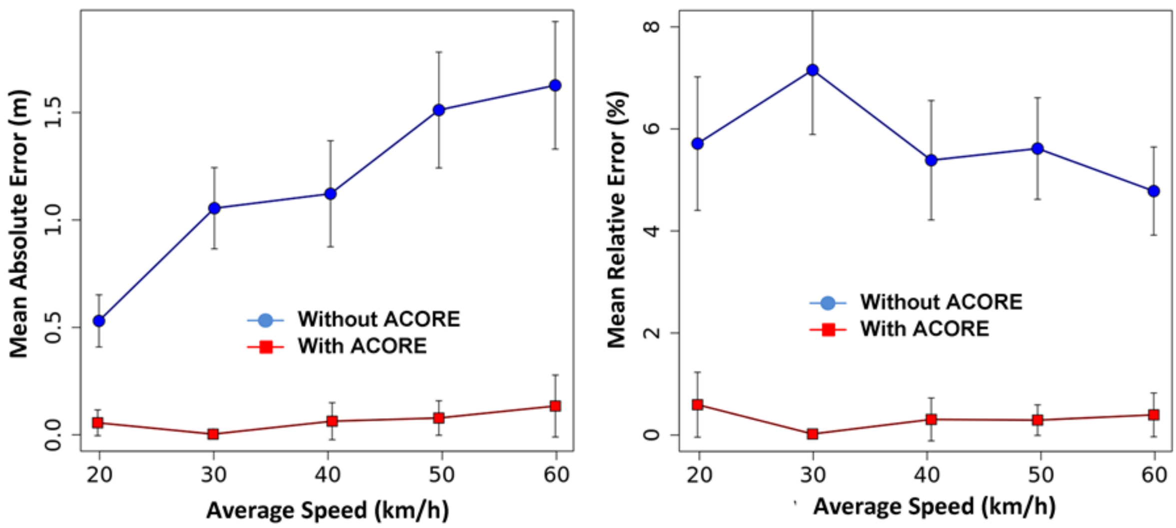


a) Without ACORE **b) With ACORE**

Figure 8. Safe Braking Distance versus Warning Braking Distance at 40 km/h.

The analysis of the mean absolute error as a function of speed (Figure 9a), for the application without ACORE, reveals a growing trend, as expected due to the relationship of direct proportionality between the braking distance and speed, reaching a value of 1,5 m at speed of 60 km/h. Applying ACORE, the average absolute error was as low as 15 cm, confirming the algorithm efficiency, regardless of speed.

Figure 9b 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 the ACORE reduced the mean relative error rate below to 1% in all speeds. The figure also shows that a greater mean relative error corresponds to occurrence of greater dispersion of values from the mean value, which occurs at speeds of 20 and 60 km/h due to errors not corrected by the ACORE.



a) Mean Absolute Error **b) Mean Relative Error**

Figure 9. Mean Error versus Average Speed.

Although the absolute and relative average errors are of great importance to the ACORE's performance evaluation, the absolute maximum error in function of speed should also be considered for an active safety application, due to the dynamic nature of the safe braking distance computation. Table 4 compares the absolute maximum error, in meters, measured at each experiment session.

Table 4. Maximum Error versus Average Speed (m)

	20 km/h	30 km/h	40 km/h	50 km/h	60 km/h
Without ACORE	1.327	2.513	2.416	2.803	3.384
With ACORE	1.045	0.058	1.506	1.315	2.240

The significant maximum average error drop forced by ACORE in the 30 km/h session was caused by correction of all relevant errors, proving the excellent performance of the algorithm in this session. It is important to register there was no loss of messages or messages received out of order in any of the sessions.

Conclusion and Future Work

In this paper, a Forward Collision Warning Application performance using WAVE architecture compatible equipment with an error correction algorithm named ACORE was presented. The results, based on the metrics, showed that without ACORE the efficiency and reliability of the application, would be compromised. The results also confirmed the ACORE's capability to allow use of GPS as a sensing device for active safety applications. The use of GPS as safe brake distance measuring device brings the benefit of avoid deployment of sensing equipment in addition to the OBUs. The average relative error margin below of 1% achieved by the ACORE, ensures reliability to the Forward Collision Warning Application using the WAVE/DSRC technology.

As future work, new experiments will be carried out with GPS update rate of 10 Hz to evaluate and improve the ACORE performance. Based on collected data from experiments, 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 objective is to carry out experiments under adverse weather conditions, absence of line of sight and GPS signal loss in scenarios such as forest areas and tunnels.

Acknowledgements

This work was partially funded by IMI project: Intelligent Mobile Infrastructure based on Cellular and Vehicular Networks - FAPERJ, by CNPq and CAPES. It was also sponsored by FAPERO and Federal Institute of Education, Science and Technology of Rondonia - IFRO.

References

1. IEEE (2015). *IEEE guide for wireless access in vehicular environments (WAVE) architecture*. IEEE Vehicular Technology Society. Available at http://ieeexplore.ieee.org, Accessed Jan. 2016.

2. Zang, T. and Delgrossi, L. (2012). *Vehicle Safety Communications: Protocols, Security, and Privacy*. Information and Communication Technology Series. Wiley.

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

4. Khainar V. and Kotecha, K. (2013). *Performance of Vehicle-to-Vehicle Communication using IEEE 802.11p in Vehicular Ad-hoc Network Environment*. In International Journal of Network Security & Its Applications, v. 5, n. 2.

5. Hill, C. (2015). *ITS ePrimer Module 13: Connected Vehicles*. United States Department of Transportation. Available at https://www.pcb.its.dot.gov/eprimer/module13.aspx#dsrc, Accessed Jan. 2016.

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

7. Zinchenko, T., Tchouankem, H., Wolf, L., and Leschke, A. (2013). *Reliability analysis of vehicle-to-vehicle applications based on real world measurements*. In Proceeding of the Tenth ACM International Workshop on Vehicular Inter-networking, Systems, and Applications, VANET ’13. ACM, pp. 11-20.

8. Missener, J. and Kelley, D. (2015). *ASN.1 Specification - Official normative ASN.1 of the SAE J2735*. SAE International. Available at http://www.sae.org, Accessed Jan. 2016.

9. COHDA (2015). *Cohda Mobility MKx SDK*. COHDA Wireless Pty, ltd. Available at http://cohdawireless.com, Accessed Jan. 2016.

10. Ivis, F. (2006). *Calculating geographic distance: Concepts and methods*. Canadian Institute for Health Information. Available at http://www.lexjansen.com, Accessed Jan. 2016.

11. Kelley, D. (2015). *DSRC implementation guide*. SAE International. Available at http://www.sae.org, Accessed Jan. 2016.

12. Lei S. and Sung K. (2014). *Spectrum Requirement for Vehicle-to-Vehicle Communication for Traffic Safety*, in Vehicular Technology Conference (VTC Spring), pp.1-5.