Collision Probability Computation for Road Intersections Based on Vehicle to Infrastructure Communication

Mahmoud Shawki and M. Saeed Darweesh
Wireless Intelligent Networks Center (WINC), Nile University, Giza 12677, Egypt
School of Engineering and Applied Sciences, Nile University, Giza 12677, Egypt
Email: m.shawki@nu.edu.eg; mdarweesh@nu.edu.eg

2020 32nd International Conference on Microelectronics (ICM) | 978-1-7281-9664-0/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICM50269.2020.9331802

***Abstract*—In recent years, many probability models proposed to calculate the collision probability for each vehicle and those models used in collision avoidance algorithms and intersection management algorithms. In this paper, we introduce a method to calculate the collision probability of vehicles at an urban intersection. The proposed model uses the current position, speed, acceleration, and turning direction then each vehicle shares its required information to the roadside unit (RSU) via the Vehicle to Infrastructures (V2I). RSU can predict each vehicle's path in intersections by using the received data. By considering vehicle dimensions in our calculation, RSU will detect a possible collision point and time to collision (TTC) for moving vehicles at the intersection. Simulation results show that this model can detect collisions occurrence early, so it will decrease the probability of a collision occurs.**

***Index Terms*—Collision probability, Vehicle to Infrastructure communication, Autonomous vehicles, Collision avoidance, Intersections.**

I. Introduction

In 2018, the United States Department of Transportation reported that there are more than 3.5 million crash cases from motor vehicles, and more than 1 million people injured because of vehicle accidents [1]. The intersection has more than 50% of those numbers [1]. The European Union has the same problem. For example, according to the federal statistical office in German, there are more than 75,000 crash cases that occurred at intersections [2]. To decrease collision at the intersections, there are many research works on collision probability computation. The method in this paper will calculate collision probability for autonomous vehicles in road intersections with vehicles dimension consideration to increase the accuracy of the calculation.

A. Related work

Intersections consider one of the most dangerous parts on roads, so some researchers design systems based on vehicle vision and sensors like radar, ultrasound, and camera to decrease the collision at intersections. Those systems were designed to recognize the surrounded environment and detected pedestrians, vehicles, and non-motor-vehicles. But those systems have many limitations because of the accuracy and range limits of sensors in addition to issues that maybe happened to vehicle vision in autonomous vehicles so vehicles can not get accurate information from each other [3] [4]. Several research papers proposed Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructures (V2I) communications networks to make each vehicle can broadcast its information such as current position, speed, acceleration, and direction with other vehicles or roadside units [5]. As shown in Fig. 1, V2V communication is a communication system that lets each vehicle share its data with others. In contrast, V2I communication is a communication system that lets each vehicle share its data with the surrounding infrastructure or roadside unit (RSU). In [6], OMNeT++ and SUMO were used to build a simulation environment to evaluate the performance of V2V communication for collision avoidance at intersections, they found that V2V communication can achieve a better accuracy for the application.

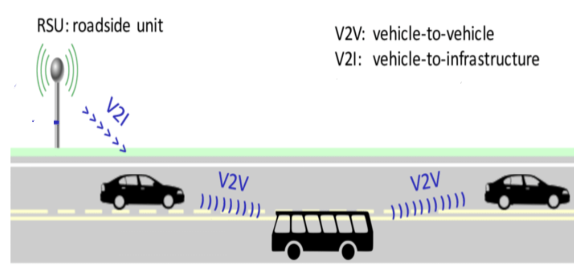

Fig. 1. V2V and V2I communications [7]

In [8], MATLAB was used to present a new method in collision avoidance of moving vehicles that introduced uncertainty trajectories based on Long-Short Term Memory networks (LSTM). The collision probability was computed in [9] and [10] by using Time-To-Collision (TTC) and Dynamic Bayesian Network (DBN) to make a decision to avoid the collision that allows the model to delay a decision for collecting more information. In [11], Trajectory prediction was used to compute the collision probability. Based on speed, position, and driver intention, this model calculated the car trajectories at the traffic intersection using least-squares fitting and used TTC to calculate the probability collision, but it did not consider the vehicle dimension in the calculation of the collision point.

B. Proposed Method

In this paper, we aim to design a system in which vehicles can transmit their information like speed, current position, and turning direction to fixed infrastructure (RSU) via V2I when they are near the intersection. When RSU receives the information, it will predict the possible trajectory for both vehicles, detect the possible collision point, and calculate TTC. Then the collision probability can be calculated by using TTC. We use the Simulation of Urban MObility (SUMO) to simulate the vehicle's motion at intersections and build motion scenarios using MATLAB/Simulink to analyze the data and compute the collision probability. All possible routes are designed for each lane in the intersection by SUMO and connect it with MATLAB by Traci4matlab library. This library is useful to use all simulation outputs in MATLAB to calculate the collision probability.

II. Methodology

When vehicles arrive to place near to the intersection, each vehicle starts to broadcast its information to RSU via V2I. Then RSU can predict the trajectories curve for both vehicles after it received current position and direction turning from them. Then possible collision points can be detected by these curves, and TTC can be computed by using their speeds. At last, the collision probability will be computed by a half-normal distribution function based on computed TTC. And we use a half-normal distribution function because TTC is always a positive number. Based on the calculated probability, RSU will start to broadcast a warning message and warn other vehicles.

A. Trajectory Prediction

The trajectory of each vehicle in intersection should be predicted to detect the collision point between two vehicles for collision probability computation. Once a vehicle is near to the intersection, RSU will receive the current position, speed, acceleration, and turning direction from each vehicle. In this section, the current position and turning direction will be used to predict the future trajectory. Each lane has four possible directions: Straight, Turn right, Turn left, and U-turn. We can consider the trajectory of turn left, turn right, and U-turn as Gaussian equation. Besides, when a vehicle moves straight, we can consider its trajectory as a straight line equation, as shown in Fig. 2.

The aim is to fit some functions to each motion trajectory for motion prediction and finding the intersection point. Then some trajectories of the vehicle's motion will be obtained at each lane from SUMO and insert them in apply curve fitting calculation to fit them with the functions. Finally, the square of the correlation coefficient is used to compare between them to estimate the goodness of fitting between the captured trajectory and function. Used functions in fitting are Gaussian function for the left turn, U-turn, and turn right and line function for straight.

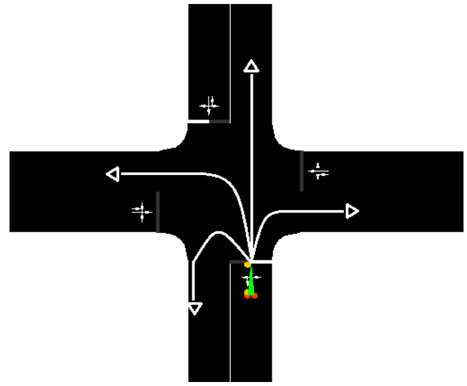

Fig. 2. Four intentions at an intersection

TABLE I
THE CORRELATION COEFFICIENT OF TWO FUNCTIONS FITTING DEGREE

Funcation \ Intention	Straight	Left turn	U-turn	Right turn
Gaussian	0.21	0.98	0.97	0.94
Line	0.99	0.32	0.35	0.44

From Table I, when the driver's intention is moving straight, the largest fitting degree is line function. The Gaussian function is the largest fitting degree when the driver's intentions are left turn, right turn, and U-turn. All of these based on the square of the correlation coefficient for evaluating the fitting degree.

So we use Eqn. (1) to predict the trajectory of vehicle motion if the driver's intention is straight and Eqn. (2) to predict the trajectory when the driver's intentions are left turn, right turn, and U-turn.

$$y = a + bx \tag{1}$$

$$y = a + e^{-(\frac{x-b}{c})^2} \tag{2}$$

B. Collision Point Calculation

After RSU predict the trajectories of both vehicles, the intersection point can be detected according to the distance between two vehicles. But first, we will represent each vehicle as a set to circles. This representation will help us consider each vehicle's dimension in the calculation and increase the number of collisions checks. Now we can check if the collision will occur or not by checking the overlapping between each set of both vehicles. The accuracy of detection is traded off with the number of circles that represent a vehicle.

So in every point in predicted trajectory, RSU will calculate the distance between each circle and the set of the opposite circles in other vehicle and the collision will occur if the distance is equal or less than the sum of diameters of both circles, as shown in Eqn. (3). While Eqn. (4) is used to determine the point of collision.

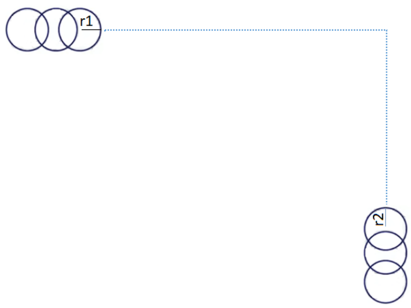

Fig. 3. Collision point calculation

$$d_{1,2} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}, r_1 + r_2 < d_{1,2} \quad (3)$$

$$c_x = \frac{(x_1 * r_2) + (x_2 * r_1)}{(r_1 + r_2)}, c_y = \frac{(y_1 * r_2) + (y_2 * r_1)}{(r_1 + r_2)} \quad (4)$$

C. Collision Probability Calculation

Now, after RSU detected the collision point, TTC will be calculated by using the received speed of each vehicle and the distance between a vehicle and the collision point. We assume TTC_1 is for the first vehicle, TTC_2 is for the second one and ΔTTC=TTC_1-TTC_2. If $\Delta TTC = 0$, the two vehicles will collide. And if $\Delta TTC \neq 0$, they won't collide. But that's for ideal case, so we put a threshold Λ to decide if they will collide or not $\Delta TTC \leq \Lambda$.

Because TTC is always a positive number, we use half-normal distribution to calculate the collision probability, as shown in Eqn. (5). Moreover, we use any of TCCs as input for half-normal distribution because Λ is very small so the difference between them is very small as well. Which Λ = 0.2 sec.

$$P(TTC) = \frac{2}{\sqrt{(2\Pi)}} e^{\frac{t^2}{2}}, \Delta TTC \leq \Lambda \quad (5)$$

III. Simulation Result

In this section, the simulation result of the proposed model will be presented to compute the collision probability. First, three scenarios will be drawn by using SUMO. In the first scenario, one vehicle turns left, and another vehicle moves straight. The second scenario is both vehicles move straight in addition to the final case is one vehicle from the left lane moves straight, and another vehicle moves left, as shown in Fig. 4, Fig. 5, and Fig. 6 representatively. Second, starting to compute collision probability by using a half-normal distribution function with zero mean and the standard deviation is equal to 1. The simulation environment is built based on two vehicles only because RSU will calculate the probability of collision between every two vehicles in intersections for large scale systems.

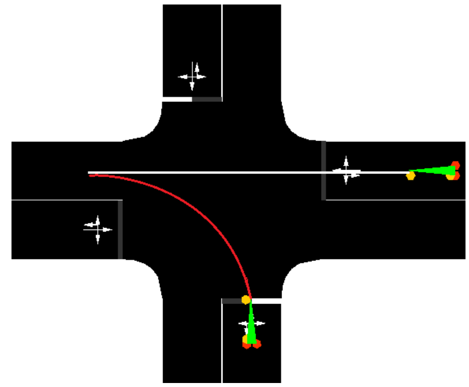

Fig. 4. Left straight to left turn

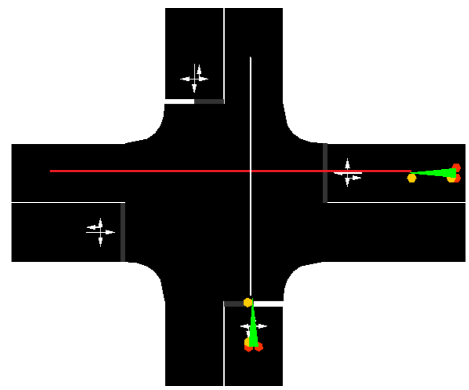

Fig. 5. Straight to straight

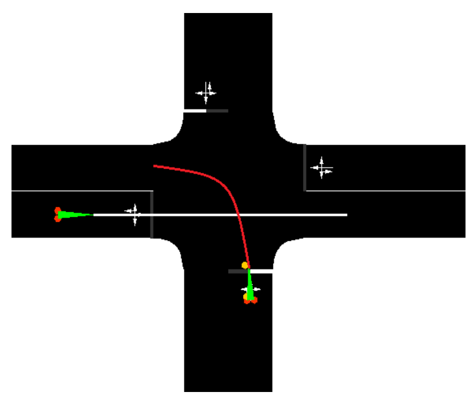

Fig. 6. Right straight to left turn

In the first scenario, the probability computation starts from 0.44, and the vehicle takes 3.5 sec to reach the maximum probability, which equals 0.8, as shown in Fig. 7. The curve in the second scenario shown in Fig. 8 is so close to the first curve, but the vehicle takes only 3 sec to reach the maximum probability, which equals 0.8. Fig. 9 shows that the third curve starts from a very high probability of 0.68 and finishes at 0.8 in just 2 sec.

As a result of dimension consideration, a collision point will be allocated in a near place for each vehicle more than the proposed method in [11]. So the distance between each

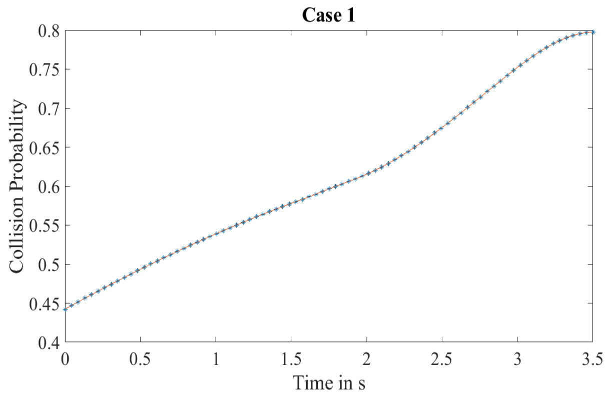

Fig. 7. Probability for left straight to left turn

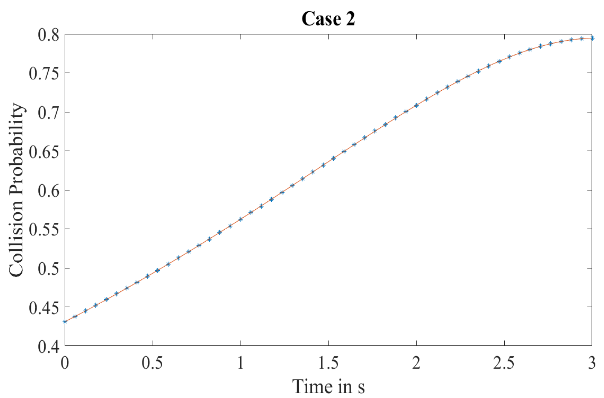

Fig. 8. Probability for straight to straight

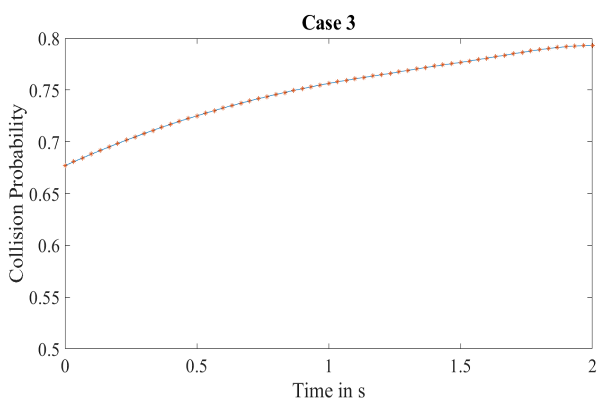

Fig. 9. Probability for right straight to left turn

vehicle and the collision point will decrease, and TTC will decrease then the probability of collision will increase. So the value of probability won't start from around zero, but it will start from higher probability. But in the third case, the collision point is very near to each vehicle, so it starts the probability calculation from a very high probability.

From Table II, the proposed method starts the collision probability computation from a high probability in the three collied cases that starts from 0.44, 0.43, and 0.68 representatively. On the other hand, the method in [11] starts its computation from a low probability that starts around 0.2 in the three cases. And in the proposed method, the maximum probability is around 0.8, and the maximum probability in [11] is around 0.9.

TABLE II
COMPARISON BETWEEN THE PROPOSED METHOD AND THE METHOD IN [11]

		Case 1	Case 2	Case 3
The Proposed Method	Start	0.44	0.43	0.68
	End	0.8	0.79	0.79
The Method in [11]	Start	0.19	0.2	0.23
	End	0.88	0.89	0.9

IV. CONCLUSION

In this paper, the aim is to compute the collision probability of vehicles at intersections. So first, the trajectories of each vehicle in the intersections are predicted according to its current position, speed, and driver's turning direction. Then we calculate the collision point and consider the vehicle's dimensions in our computation. Finally, we calculate the collision probability. The collision probability curves show that the proposed method can start its computation from high probability in collided cases by comparing it with the literature. And that will help to detect collided cases early.

REFERENCES

[1] "Motor Vehicle Safety Data — Bureau of Transportation Statistics," Bts.gov, 2020. [Online]. Available: https://www.bts.gov/content/motor-vehicle-safety-data. [Accessed: July 5, 2020].

[2] "Statistisches Bundesamt,Verkehrsunfalle," Germany 2016.

[3] J. Giacalone, L. Bourgeois and A. Ancora, "Challenges In Aggregation of Heterogeneous Sensors for Autonomous Driving Systems," in *IEEE Sensors Applications Symposium (SAS)*, 2019.

[4] F. Rosique, P. Navarro, C. Fernandez and A. Padilla, "A Systematic Review of Perception System and Simulators for Autonomous Vehicles Research," in *MDPI Sensors*, vol. 3, no. 19, p. 648, 2019.

[5] S. Benkirane and M. Benaziz, "Performance Evaluation of IEEE 802.11p and IEEE 802.16e for Vehicular Ad hoc Networks Using Simulation Tools," in *IEEE 5th International Congress on Information Science and Technology (CiSt)*, 2018.

[6] I. Rashdan, M. Schmidhammer, F. Muller and S. Sand, "Performance Evaluation of Vehicle-to-Vehicle Communication for Cooperative Collision Avoidance at Urban Intersections," in *IEEE 86th Vehicular Technology Conference (VTC-Fall)*, 2017.

[7] Ahmed Hosny, Mohamed Yousef, Wessam Gamil, Mohamed Adel, Hassan Mostafa, and M. Saeed Darweesh, "Demonstration of Forward Collision Avoidance Algorithm Based on V2V Communication," in 8th International Conference on Circuits and Systems Technologies (MOCAST), Thessaloniki, Greece, May 2019.

[8] H. Ding, H. Wu, L. Dong, and Z. Li, "Vehicle Intersection Collision Monitoring Algorithm Based on VANETs and Uncertain Trajectories," in *16th International Conference on Intelligent Transportation Systems Telecommunications (ITST)*, 2018.

[9] S. Lefevre, R. Bajcsy, and C. Laugier, "Probabilistic Decision Making for Collision Avoidance Systems: Postponing decisions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013.

[10] S. Lefevre, C. Laugier, and J. Ibanez-Guzman, "Evaluating risk at road intersections by detecting Conflicting Intentions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012.

[11] R. Zeng, W. Sheng, and D. Yang, "Collision Probability Computation Based on Vehicle-to-Vehicle Communication," in *IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, 2015.