Vehicle Intersection Collision Monitoring
Algorithm Based on VANETs and Uncertain Trajectory

Hongwei Ding, Hao Wu, Lan Dong and Zejun Li
Vehicle Intersection Collision Monitoring Algorithm Based on VANETs and Uncertain Trajectory

1st Hongwei Ding  
State Key Laboratory of Rail Traffic Control and Safety  
Beijing Jiaotong University  
Beijing, China  
16120055@bjtu.edu.cn

2nd Hao Wu, Member IEEE  
State Key Laboratory of Rail Traffic Control and Safety  
Beijing Jiaotong University  
Beijing, China  
hwu@bjtu.edu.cn

3rd Lan Dong  
School of Computer and Information Technology  
Beijing Jiaotong University  
Beijing, China  
13211035@bjtu.edu.cn

4th Zejun Li  
State Key Laboratory of Rail Traffic Control and Safety  
Beijing Jiaotong University  
Beijing, China  
1720091@bjtu.edu.cn

Abstract—In order to ensure driving safely, the driving safety assistance system must be able to aware of potential collision accidents in advance, especially significant for the intersection where traffic accidents occur more frequently. Considering that VANETs is one of the most important applications for improving the safety of driving, furthermore, vehicles have an inherent uncertainty of location because the exact position of a moving object is known, with certainty, only at the time of an update on position information. In order to reduce the accident rate at intersection and combine the driving characteristics of vehicles at traffic intersections, a vehicle intersection collision monitoring algorithm based on VANETs and uncertain trajectory is proposed. The algorithm is divided into two categories: uncertain trajectory prediction algorithm and vehicle collision monitoring algorithm. The proposed approach provides approximate answers to the user at the users required level of accuracy while achieving near-optimal communication and computational costs. Finally, extensive experiments were conducted to show the efficiency and efficacy of the proposed approach.

Index Terms—collision monitor, intersection, uncertain trajectory, VANETs

I. INTRODUCTION

Collision avoidance is one of the most critical concerns in traffic safety, and it is becoming increasingly important as traffic volume increases. Recently, with changes in social needs and automotive technology, autonomous driving has become an important concern. Collision avoidance system has been becoming a significant component in the current autonomous driving research to ensure driving safety[1].

According to traffic accident statistics from the Road Traffic Authority, approximately 75% of fatal traffic accidents resulted in collisions among vehicles in recent years[2].In particular, more than 40% of all crashes causing injuries or fatalities worldwide occur in intersection. Therefore, a number of solutions have been proposed to mitigate or avoid collision[3].

Collision avoidance systems provide a service based on the locations of moving objects; therefore, the accuracy of the location information has a direct influence on the service quality of the system. The key factor of system quality is to know the exact present position of a vehicle, and to predict its future position accurately by monitoring the movement of a vehicle for collision avoidance [4] [5].

Positioning systems utilizing technology such as a GPS device in a vehicle and roadside sensors can provide location samples only at discrete time instants. Thus, the location of a moving object is never definite between two consecutive samples. Because a vehicle moves continuously, the moving information such as position, velocity, and direction of a vehicle changes constantly[6].

Therefore, this paper introduces a collision monitor model considering the location uncertainty of vehicles in the intersection to increase the accuracy of collision risk detection. The primary contributions of this paper are summarized as follows:

• This paper introduces a novel uncertain trajectory prediction method, which combined with uncertain position and the main line of trajectory.

• This paper introduce a novel collision monitor algorithm which which predicts collision possibilities based on VANETs and the uncertain trajectory.

• This study addresses this issue by proposing a method that achieves minimal use of wireless network bandwidth and optimal computational cost while not missing true warnings.

• A series of simulations are conducted to demonstrate that the proposed algorithm markedly outperforms a conventional solution in terms of reducing collision risks and the computational consumption.

The remainder of this paper is organized as follows: Section II discusses related work, Section III details the system model, Section IV proposed approach for collision avoidance monitoring algorithm, Section V presents and analyze simulation results, and Section VI concludes this paper.

II. RELATED WORK

In recent years, significant research and development activities have been performed to avoid collisions. When a vehicle predicts collisions within a black zone, it has one to many relationships with other vehicles. By
evaluating information independently, the objects also assess
the situation independently and make separate decisions on
whether the situation is critical or not, and different judgments
cause traffic congestion and conflicts[7]. To resolve these
problems, a supervisory role that manages the overall situation
of a black zone is necessary. Therefore, this study uses a
centralized approach to solve priority and information conflicts
for improving safety.

Many proposed collision-predicting methods assume that
position data of moving vehicles are precise, and the future
positions of vehicles are predicted based on periodically
updated information such as position and velocity at every
time instant (e.g., 0.1 s). However, these approaches can cause
undesirable consequences such as missing a true warning and
poor response times. In addition, in real traffic scenarios,
thousands of vehicles operate in the same district at the same
time. It is unlikely for a server to fulfill the task of cooperative
localization owing to limited computational resources as well
as limited network bandwidth[5] [8]. This study addresses this
issue by proposing a method that achieves minimal use of
wireless network bandwidth and optimal computational cost
while not missing true warnings.

III. SYSTEM MODEL

A. Cooperative vehicle infrastructure system within an inter-
section

VANETs are created by applying the principles of mobile
ad hoc networks. It is the spontaneous creation of a wireless
network for data exchange to the domain of vehicles. VANETs
consist of a set of moving objects (e.g., vehicles, pedestrian
with mobile devices) and servers (e.g., RSUs). In VANETs,
V2I communication, communication between vehicles and a
server of roadsides, corresponds to a server-clients communica-
tion in mobile environments. This study uses a server-centric
approach based on the cooperation of objects passing through
an intersection and a server[9]. The intersection mark points
can help determine whether vehicles are in the intersection.

![Figure 1. Vehicle cooperation communication system](image)

In this paper, a report is a message transferred from a
target vehicle to a server. It is assumed that moving objects
$O_1$ through $O_{10}$ are vehicles, and $O_i$ is a target vehicle that
communicates with a server for collision monitor. In addition,
moving objects $O_1$ through $O_{10}$ can be the target vehicle $O_i$.
A server monitors target vehicle $O_i$ and determines if there is
a collision, based on the information from $O_i$ and that of its
peers (i.e., $O_1$-$O_{10}$) within the intersection area. When a moving
vehicles $O_i$ approaching the intersection. The server predicts
the uncertain trajectory of the target vehicle $O_i$ and monitors
collision probability with other vehicles in the intersection,
based on the reports from other vehicles[10]. The server then
sends the target vehicle the first warning if the collision
probability is higher than the desired accuracy of $O_i$. The target
vehicle $O_i$, based on the warning and notification messages
received from the server, may take action to avoid collisions
(e.g., speed adjustment within the safe reaction distance).

B. Uncertainty region and report delivery

![Figure 2. Uncertainty position of $O_i$](image)

This study assumes that a moving vehicle is located any-
where in the uncertainty region at a given point in time. Fig.2
shows the location uncertainty of vehicle $o_i$ at time $t$. Target vehicle $o_i$ is denoted by the point, and its uncertainty region is denoted by a solid-line circle $U_i$ with center $C_i$, and radius $r_i$. The smaller the difference between the expected location and the actual location, the smaller the radius of the circle representing the uncertainty region.

A report $rpt$ consists of six attributes, $<rpt = o_i, C_i, V_i, r_i, t_i, \delta_{\text{min}}>$ Where $o_i$ indicates the target object, $C_i$ and $V_i$ indicate the center position $(x, y)$ and velocity $(v_x, v_y)$, respectively, of target object $o_i$ at time stamp $t_i$, and $r_i$ is radius of the uncertain area. $\delta_{\text{min}}$ is the accuracy of the approximate answer, which reflects the users required level of accuracy. A report is generated when a vehicle deviates from its uncertainty region, which is sent to the server.

C. Collision risk considering uncertain trajectory within a dangerous area

A moving object $o_i$ may have a probability of collision with any other objects that appear in the black zone between the times when the target object enters and exits the black zone. Let $t_b$ and $t_e$ be the times when target object $o_i$ enters and leaves the black zone, respectively. The movements of $o_i$ within a black zone depend on the number of vehicles in the black zone and the nature of the black zone such as layout or traffic pattern. Assume that $B = o_1, o_2, ..., o_n$ is a subset of $O$, which has moving objects appearing within the black zone for the time interval $[t_b, t_e]$, and they have a collision probability with $o_i$. Let $T_{ri}$ be an uncertain trajectory of moving object $o_i$ and $S = T_{r1}, T_{r2}, ..., T_{rn}$ be a set of the uncertain trajectories of moving objects.

Consider the scenario depicted in Fig.4, which illustrates four trajectories: $T_{r1}, T_{r2}, T_{r3}$, and $T_{r4}$, which are based on the initial positions and velocity of object $o_1, o_2, o_3$, and $o_i$, respectively. The uncertain trajectories run through the time interval $[t_b, t_e]$. Clearly, a moving object that does not have an overlapping area with uncertain trajectories of other objects has little collision probability. The shaded parts in the sheared oval cylinders have a larger collision probability between $o_i$ and one of the other objects $o_1, o_2, o_3$, if considering location. Ignoring location uncertainty, the nonzero collision risk neighbors of $T_{ri}$ are only $T_{r2}$ at $t_2$. However, if location uncertainty is considered, $T_{ri}$ has collision risk possibilities with $T_{r2}$ and $T_{r1}$ within $[t_1, t_3]$ and $[t_4, t_5]$, respectively, as well as at $t_2$. In addition, if only consider the overlap of trajectories, $T_{r3}$ has no collision risk with $T_{r3}$ in $[t_b, t_e]$.

IV. METHODOLOGY

A. Uncertain trajectory prediction algorithm for vehicles

According to the driving intention of the vehicle at the intersection, we divide the trajectory of the vehicle into two types: going straight and turning. For determining the range of uncertain trajectory for vehicles, we need to calculate the main line equation of the trajectory firstly. Then according to the trajectory main line equation and the definition of uncertain position, we can get the range of uncertain trajectory of the vehicle. The following is a brief introduction of the solution to the main line of uncertain trajectory when going straight and turning.

**going straight**: The linear expression of the trajectory main line when the vehicle going straight can be determined in combination with the current position $(x_0, y_0)$ and the rotation angle $\phi$ of the front wheel of the vehicle.

$$ y - y_0 = \frac{\phi}{90} (x - x_0) $$

**turning**: Because the trajectory curve of the vehicle and the curvature of the curve are continuous, we describe the vehicle’s trajectory using cubic curve interpolation. Each segment of the Vehicle’s trajectory can use a cubic equation to describe.

$$ S_i(x) = a_i + b_i(x-x_i) + c_i(x-x_i)^2 + d_i(x-x_i)^3, \quad i = 1, 2, \cdots, n-1 $$

Where coefficient $a_i, b_i, c_i, d_i$ can be determined by the properties of the cubic curve, $x_i$ is the abscissa of the vehicle’s location.

An uncertain trajectory $T_{ri}$ is a path consisting of all uncertainty regions of moving object $o_i$ in a time interval. In Fig. 3, it is assumed that moving object $o_i$ moves along a straight line with a constant speed, which is based on the velocity $(v_x, v_y)$ at current time $(t_0)$. Thus, the expected location of $o_i$ at time $t$ is evaluated under the assumption that it is based on the current velocity $(v_x, v_y)$ of $o_i$ with acceleration $a = 0$.

Let $L(o_i, t)$ be the expected location of vehicle $o_i$ at time $t$, $(x_{i0}, y_{i0})$ and $(v_x, v_y)$ denote the location and velocity, respectively, of $o_i$ at current time $(t_0)$. Then, the expected location $L(o_i, t)$ of $o_i$ at time instant $(t_k)$ is given by:

$$ L(o_i, t_k) = (x_{ik}, y_{ik}) = (x_{i0} + v_x(t_k - t_0), y_{i0} + v_y(t_k - t_0)) $$

**Fig. 3.** Uncertain trajectory of $o_i$

B. Collision probability between moving objects

This section computes the collision probability mathematically. Assume that target vehicle $o_i$ and the other vehicle $o_j$ are located anywhere in uncertainty region $U_i$ and $U_j$ at time $t$. If at time $t$ the uncertainty location $U_i$ of target vehicle $o_i$ has an intersection area $(C)$ with uncertainty location $U_j$ of another vehicle $o_j$, then there is a possibility of collision. Fig. 5 illustrates the computation of the collision risk probability of $o_i$ and $o_j$ at time $t$ when both $o_i$ and $o_j$ are located in the red area $(C)$. As shown in the figure, $p_c = \frac{area(C)}{area(t_{ij})} \times \frac{area(C)}{area(t_{ij})}$

**Fig. 6** considers there cases separately: $(a)$ $\text{dist}(c_i, c_j, t) > \left| r_i - r_j \right|$, $(b)$ $\text{dist}(c_i, c_j, t) < \left| r_i - r_j \right|$ and $(c)$ $\left| r_i - r_j \right| < $
The algorithm consists of filter and refinement steps. The first step determines the time interval \([t_b, t_e]\) of \(o_i\) based on the information of new report \(rpt\). Then, it decides \(B\), which is a set of objects existing within the intersection area for the time interval \([t_b, t_e]\). Next, to calculate the overlap area \(C\) between the trajectory \(T_{rj}\) of \(o_j\) and the trajectory \(T_{rj}\) of the other vehicles \(o_j\). Finally, the refinement step computes the collision probability with \(o_i\) for each candidate \(o_j\). This step determines the answer set \(A\) satisfying \(P_{cij} > \delta_{min}\). In addition, in order to avoid too many early and inaccuracy warning message, the answer set should satisfy \(t - t_0 \leq 2s\) as well, where \(t\) is the collision time, \(t_0\) is the current time.

### Algorithm 1: Vehicle Collision Monitoring Algorithm

**Input:** \(rpt\) new report  
**Output:** \(A\): an approximate answer set with a required accuracy level

1. while receiving a \(rpt\) do  
2. for the \(rpt\) from \(o_i\) do  
3. if \(o_i \notin O\) then  
4. put \(o_i\) into \(O\);  
5. determine \([t_b, t_e]\) of \(o_i\);  
6. decide \(B\) as a set of objects existing in the intersection during \([t_b, t_e]\) of \(o_i\);  
7. for each peer \(o_j \in B\) do  
8. if \(C \leq 0\) then  
9. CONTINUE  
10. else  
11. if time interval \(I\) is not included in \([t_b, t_e]\) then  
12. CONTINUE  
13. else  
14. if \(P_{cij} > \delta_{min}\) \(\bigcap t - t_0 \leq 2s\) then  
15. \(o_j \in A \leftarrow o_j\);  
16. send \(A\) as an answer set to \(o_i\);  
17. end if  
18. end if  
19. end if  
20. end for  
21. end if  
22. end for  
23. end while

### C. Monitoring algorithm for collision avoidance

Naturally, the server administering a dangerous area has all relevant information such as the location and layout of the area. The server has received the reports \(rpt\) from the vehicles which are going to drive into the intersection. Each vehicle is not only a target vehicle but also another vehicle for each target vehicle within the monitored intersection.

**Algorithm 1** provides the detailed steps of the monitoring algorithm for collision avoidance. The algorithm receives a new report \(rpt\) from a target vehicle in the monitored intersection and returns the approximate answer \(A\) (of collision candidates) satisfying the minimum required accuracy \(\delta_{min}\). The algorithm consists of filter and refinement steps. The first step filters out any vehicles that has no collision probability with \(o_j\). The refinement step computes the collision probability with \(o_i\) for each other vehicles \(o_j\) which remain after the filter step. It returns the answer set \(A\) (of collision candidates) satisfying \(\delta_{min}\). When a new report arrives, the server investigates whether this update affects the query result.

The new report \(rpt\) satisfies the following condition: (1) the report comes from a new vehicle, or (2) the object \(o_i\) of set \(O\) deviates from the uncertainty region. First, the filter step should determine the time interval \([t_b, t_e]\) of \(o_i\) based on the information of new report \(rpt\). Then, it decides \(B\), which is a set of objects existing within the intersection area for the time interval \([t_b, t_e]\). Next, to calculate the overlap area \(C\) between the trajectory \(T_{rj}\) of \(o_i\) and the trajectory \(T_{rj}\) of the other vehicles \(o_j\). Finally, the refinement step computes the collision probability with \(o_i\) for each candidate \(o_j\). This step determines the answer set \(A\) satisfying \(P_{cij} > \delta_{min}\). In addition, in order to avoid too many early and inaccuracy warning message, the answer set should satisfy \(t - t_0 \leq 2s\) as well, where \(t\) is the collision time, \(t_0\) is the current time.

### Algorithm 1: Vehicle Collision Monitoring Algorithm

**Input:** \(rpt\) new report  
**Output:** \(A\): an approximate answer set with a required accuracy level

1. while receiving a \(rpt\) do  
2. for the \(rpt\) from \(o_i\) do  
3. if \(o_i \notin O\) then  
4. put \(o_i\) into \(O\);  
5. determine \([t_b, t_e]\) of \(o_i\);  
6. decide \(B\) as a set of objects existing in the intersection during \([t_b, t_e]\) of \(o_i\);  
7. for each peer \(o_j \in B\) do  
8. if \(C \leq 0\) then  
9. CONTINUE  
10. else  
11. if time interval \(I\) is not included in \([t_b, t_e]\) then  
12. CONTINUE  
13. else  
14. if \(P_{cij} > \delta_{min}\) \(\bigcap t - t_0 \leq 2s\) then  
15. \(o_j \in A \leftarrow o_j\);  
16. send \(A\) as an answer set to \(o_i\);  
17. end if  
18. end if  
19. end if  
20. end for  
21. end if  
22. end for  
23. end while

### V. Experiment and Analysis

This section evaluates the performance of CAMA using three metrics: (1) the communication cost, which measures the total number of messages transferred between vehicles and a server administering an intersection; (2) the computational cost, which measures the query processing time of a message per minute; and (3) the quality of the approximate query
answer, which can reflect the accuracy of collision avoidance monitor algorithm.

TABLE II
SIMULATION PARAMETER SETTINGS.

| Parameter                        | Range                        |
|----------------------------------|------------------------------|
| simulation environment           | MATLAB_R2014a                |
| the size of intersection         | 20m x 20m                    |
| traffic flow density (vehicles/line/hour) | 600, 800, 1000, 1200, 1400, |
| the average speed of vehicle     | 2, 3, 4, 5, 6, 7, 8, 9, 10m/s|
| number of lanes                  | 4 (no signal lights)         |
| radius of uncertainty region     | 3(m)                         |
| ratio of vehicles that deviate from | 10 (%)                      |
| the driving attention of vehicles| go straight, turning left, turning right |
| message size                     | 128 bytes                    |
| beacon interval                  | 0.1 (sec)                    |
| number of query issuer (vehicles)| 10, 20, 30, 40, 50           |
| the accuracy of approximate answer ($\delta_{\text{min}}$) | 0.20                         |

Finally, Table II summarizes the parameters and relevant values used in the simulations. Each simulation was conducted with a variety of ranges for a single parameter, while keeping the other parameters at the default values which are shown in bold in Table II.

As shown in Figure 5, the accuracy of the collision detection changes as the traffic volume increases. The collision detection accuracy rate becomes smaller as the traffic volume increases. This is because the traffic volume increases, and the trajectories of different vehicles are different. There is always a part of the collision that will be missed. The baseline is the vehicle collision prediction method proposed in [11]. Our proposed collision monitoring algorithm CAMA is obviously better than the baseline method in accuracy of the collision detection, especially when the traffic volume increases continuously.

Fig. 5 shows the number of transmitted messages as a function of the number of vehicles as query issuers. For the baseline method, the number of transmitted messages increases linearly with the value of query vehicles, because the number of messages increases typically with the value of query vehicles.

Fig. 6 and Fig. 7 shows the comparison of the results from CAMA and the baseline method in terms of the number of transmitted messages. Both of them have two types of the y-axis, because the value of the difference between CAMA and the baseline method is greater. The left y-axis is for CAMA, and the right y-axis is for the baseline method.

Fig. 6 shows the number of transmitted messages as a function of the vehicle speed. The number of transmitted messages on the baseline method are constant regardless of the objects speed, whereas CAMA shows a marginal reduction based on the vehicle speed. This is plausible in CAMA since the time interval in intersection of the query issuer decreases as the value of vehicle speed increases. With regard to the baseline method, messages are transmitted to the server every 0.1 s.

Fig. 7 shows that, the time interval in the intersection of the query vehicles decreases as the speed of vehicle increases. Fig. 9 show that, the query processing time of both CAMA and the baseline method increases with the number of query vehicles increase.
The shorter query processing time accorded to CAMA is a critical issue in vehicular ad hoc networks because the query processing of a server can cause a bottleneck in the system. As shown in Fig.10, the query processing times of CAMA and the baseline method decrease slightly as the value of vehicle speed increases. This is because, as shown in Fig.8, the time interval in the intersection of the query issuer decreases as the average speed of vehicle increases. Fig.11 show that, the query processing time of both CAMA and the baseline method increases with the number of query vehicles increase.

VI. CONCLUSION

This paper proposed a probabilistic approach called CAMA, a monitoring algorithm for collision avoidance of moving vehicles within an intersection in VANETs. To this end, this paper introduced an uncertainty region, which saves computational and communication costs. By means of a series of simulations, it has also shown that the performance of CAMA is superior to that of the existing solution in terms of communication and computational costs. As future work, a method to minimize false alarms and recommend actions to avoid collision after a warning are planned under the proposed approach for collision avoidance.

ACKNOWLEDGMENTS

This work was partly supported by the State Key Laboratory of Rail Traffic Control and Safety Contract No. RCS2016ZT015, Beijing Jiaotong University, the Fundamental Research Funds for the Central Universities under Grant 2017JBM312, the NFSC Outstanding Youth under Grant 61725101.

REFERENCES

[1] V. Sotoudehnejad and M. R. Kermani, “Velocity-based variable thresholds for improving collision detection in manipulators,” in 2014 IEEE International Conference on Robotics and Automation (ICRA), May 2014, pp. 3364–3369.
[2] X. Li, L. Li, F. Flohr, J. Wang, H. Xiong, M. Bernhard, S. Pan, D. M. Gavrila, and K. Li, “A unified framework for concurrent pedestrian and cyclist detection,” IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 2, pp. 269–281, Feb 2017.

[3] J. A. Douthwaite, A. D. Freitas, and L. S. Mihaylova, “An interval approach to multiple unmanned aerial vehicle collision avoidance,” in 2017 Sensor Data Fusion: Trends, Solutions, Applications (SDF), Oct 2017, pp. 1–8.

[4] Z. Boukhers, K. Shirahama, and M. Grzegorzek, “Example-based 3D trajectory extraction of objects from 2D videos,” IEEE Transactions on Circuits and Systems for Video Technology, vol. PP, no. 99, pp. 1–1, 2017.

[5] M. Brnnstrm, F. Sandblom, and L. Hammarstrand, “A probabilistic framework for decision-making in collision avoidance systems,” IEEE Transactions on Intelligent Transportation Systems, vol. 14, no. 2, pp. 637–648, June 2013.

[6] C. S. Lin and S. Y. Wu, “Processing directional continuous range queries for mobile objects on road networks,” in The 4th Annual IEEE International Conference on Cyber Technology in Automation, Control and Intelligent, June 2014, pp. 330–335.

[7] B. Xu, O. Wolfson, and H. J. Cho, “Monitoring neighboring vehicles for safety via V2V communication,” in Proceedings of 2011 IEEE International Conference on Vehicular Electronics and Safety, July 2011, pp. 280–285.

[8] K. Narukawa, T. Yoshiike, K. Tanaka, and M. Kuroda, “Real-time collision detection based on one class SVM for safe movement of humanoid robot,” in 2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids), Nov 2017, pp. 791–796.

[9] C. Wei, W. Xinye, and H. Hongwu, “Anti-rear-end collision safe running and its warning of vehicle on highway,” in 2016 Eighth International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), March 2016, pp. 97–101.

[10] K. Driggs-Campbell, V. Govindarajan, and R. Bajcsy, “Integrating intuitive driver models in autonomous planning for interactive maneuvers,” IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 12, pp. 3461–3472, Dec 2017.

[11] P. Wang and C. Y. Chan, “Vehicle collision prediction at intersections based on comparison of minimal distance between vehicles and dynamic thresholds,” vol. 11, no. 10, 2017, pp. 676–684.