2015 IEEE Intelligent Vehicles Symposium (IV)
June 28 - July 1, 2015. COEX, Seoul, Korea

Threat Prediction Algorithm based on Local Path Candidates and Surrounding Vehicle Trajectory Predictions for Automated Driving Vehicles

Jae-Hwan Kim and Dong-Suk Kum†

Abstract— **Among others, a reliable threat prediction algorithm is one of the key enabling technologies for the commercialization of the automated driving systems and other driver assistance systems. Previous algorithms that use Time-to-Collision (TTC) as a measure of threat tend to assume constant state and constant input; e.g. constant yaw rate and constant acceleration. Although the predictability of these algorithms is acceptable within a one second time horizon, it becomes invalid for predictions over one second because yaw rate and acceleration are highly unlikely to be constant. Therefore, in this paper, we propose a threat prediction algorithm that can accurately predict TTC over a longer time horizon based on future trajectory predictions of a surrounding vehicle. First, a comprehensive set of local path candidates is generated along the curvilinear coordinates using a quintic (5^{th} order) polynomial with respect to the arc-length corresponding to the different lateral offsets. Trajectory prediction of a surrounding vehicle is accomplished by introducing target lane detection, which is estimated according to the amount of difference between the current motion and the centerline of the driving lane. Based on these future vehicle trajectories, TTC is computed by comparing the entrance and exit time of two vehicles into and out of the conflict area where the occupied spaces of two vehicles overlap. Finally, in order to provide threat assessment results, the inverse TTC values obtained above are plotted on a 2-dimensional trajectory plane where each set of the tangential acceleration and the initial yaw acceleration values represents each local path candidate. Thus, these threat assessment results can be directly utilized to determine a driving strategy of autonomous vehicles.**

I. Introduction

Accelerated by the Grand/Urban Challenge, hosted by DARPA to promote research on unmanned military systems, automated driving vehicles have been the subject of active research over the last decade. Global automotive companies such as GM, BMW, Volvo, and Mercedes-Benz have recently released announcements about the commercializing of automated driving vehicles by 2020. However, in order for the automated driving vehicles to be commercialized, safety and reliability must be guaranteed so that societies can comfortably embrace this potentially intrusive technology. Among others, reliable threat prediction algorithm is the key enabling technology to realize safe and reliable automated driving systems that can cope with high risk situations.

The existing prediction methods to assess risks associated with potential accidents can be classified into two broad categories: quantitative risk evaluation and binary collision prediction. The first focuses on quantifying risks as one indicator using the current state of the vehicle, such as position, heading, or velocity. Several types of quantitative risk evaluation to avoid collision with vehicle capability [1] or calculation of collision probability [2] can be seen in the literature. Among others, Time-to-Collision (TTC) is the most representative indicator because this time-based indicator physically presents both spatial proximity and speed difference [3]. Generally, using the current state of vehicles, TTC can be derived by calculating the time until the relative distance between two vehicles becomes zero. However, since the idea that future behavior of a vehicle will not change is inherently assumed in this method, eventually, above method to compute TTC is equal to introduce the trajectory prediction based on the constant kinematic model (e.g. Constant Velocity (CV) and Constant Yaw Rate and Acceleration (CYRA) model). When these models are applied, the difference between the predicted trajectory and the realistic driving situation becomes sharply large after one second because the velocity, yaw rate, and acceleration are highly likely to change after one second [4][5]. In other words, these methods cannot detect true TTC that is larger than one second, and thus, are not

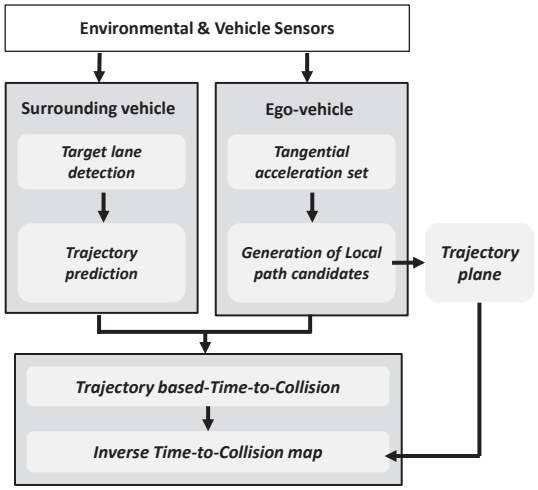

Figure 1. Framework of the proposed threat prediction algorithm

J. H. Kim is with the Cho Chun Shik Graduate School for Green Transportation, KAIST, Daejeon 350-701, Republic of Korea (corresponding author to provide phone: +82-42-450-1286; e-mail: kimjh2608@ kaist.ac.kr).

D. S. Kum is with the Cho Chun Shik Graduate School for Green Transportation, KAIST, Daejeon 350-701, Republic of Korea (corresponding author to provide phone: +82-42-450-1250; fax: +82-42-350-1266; e-mail: dskum@ kaist.ac.kr).

†:corresponding author

reliable for the assessment of risks that can evolve in the future. On the other side of prediction methods, binary collision prediction adopting the motion planning method was proposed to detect whether a collision will occur between two vehicles [6][7]. However, since this approach completely eliminates local path candidates that overlap with those of surrounding vehicles, it is not possible to determine a safety margin among them. Especially, if the number of surrounding vehicles increases on the road, the number of the available safe trajectories will rapidly vanish, and thus, the decision making process cannot be completed properly.

Therefore, in this paper, we propose the threat prediction algorithm that can accurately generate TTC over a longer time horizon based on both local path candidates and surrounding vehicle trajectory prediction as shown in Fig 1. Conceptually, the proposed approach aims to take the strength and complement the weaknesses of the above two methods. Specifically, the prediction horizon is extended beyond one second by applying the trajectory prediction, and then, safety margin is estimated denoted by inverse TTC (TTC^{-1}) at each local path candidate. The remainder of this paper is organized as follows. Section II describes the generation of local path candidates for the ego-vehicle. Section III presents the trajectory prediction of a surrounding vehicle based on target lane detection. In Section IV, we suggest the TTC for calculating the safety margin. In order to calculate the TTC, we compare the arrival time of each vehicle to four vertices of the conflict area where two vehicles overlap. Simulation results are shown in Section V; these results represent inverse TTC values of each local path candidate on the trajectory plane. Conclusion is discussed in Section VI.

II. GENERATION OF LOCAL PATH CANDIDATES

To predict the risks inherent in surrounding vehicles, local path candidates are generated that the ego-vehicle can follow within the road boundary in the near future. Here, state space sampling method is adopted for the trajectory generation process [8]. The state space sampling method generates trajectories corresponding to the set of lateral offsets aligned with the reference path. In contrast to the control space sampling method, it can generate local trajectories that do not violate the road geometry. Thus, the state space sampling method is suitable for generation of local trajectories.

A. Quintic (5^{th} order) polynomial in curvilinear coordinates

A quintic polynomial with respect to the arc-length is introduced as a basis function for generation of local path candidates with different lateral offsets. Instead of designing trajectories in Cartesian coordinates, we generate local path candidates in curvilinear coordinates in order to take the curvature of the road into account; where the horizontal axis is the arc-length, (s), and the vertical axis is the lateral offset, (q). Existing methods to generate local paths in curvilinear coordinates often adopt the quintic polynomial with respect to time for both the arc-length and the lateral offset, ($s(t)$) and ($q(t)$), as a basis function [9]. In this study, however, the lateral offset is described by a quintic polynomial function with respect to the arc-length, ($q(s)$), which allows to conveniently compute the yaw rate of all local path candidates, ($\omega(t)$) (Section II.B). Note that there exists a unique initial yaw acceleration corresponding to each local path candidate due to the lateral offsets, and thus the initial yaw acceleration is selected to represent the vertical component of the trajectory plane.

In terms of the generation of local path candidates, a quintic polynomial function is necessary to be able to cover any initial yaw rate and curvature values for the given set of final lateral offsets. Otherwise, a set of local path candidates that satisfy both the initial and final constraints cannot be generated. For instance, previously proposed methods using a cubic polynomial [10] or sigmoid function [11] cannot meet this requirement. Furthermore, in terms of the driving comfort, this function was proved as optimal trajectory with minimizing squared jerk as shown in [7]. Also, the arc-length that is the moving distance along the centerline of the current lane is modeled as a quadratic (2^{nd} order) polynomial regarding time.

$$q(s) = \begin{cases} a_5 s^5 + a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0 & (0 \le s < s_f) \\ q_f & (s \ge s_f) \end{cases} \qquad (1)$$

$$s(t) = s_0 + v_T t + \frac{1}{2} a_T t^2$$

The boundary conditions that are necessary to determine the coefficients of a lateral offset, ($q(s)$) are shown in (2).

$$\begin{array}{lllll} q(0) = q_i & \dfrac{dq(0)}{ds} = \tan\psi & \dfrac{d^2q(0)}{ds^2} = \left(\dfrac{\omega}{v_T} - k_c\right) \cdot \left[1 + \left(\dfrac{dq}{ds}\right)^2\right]^{3/2} \\ q(s_f) = q_i & \dfrac{dq(s_f)}{ds} = 0 & \dfrac{d^2q(s_f)}{ds^2} = 0 \end{array} \qquad (2)$$

Note that ψ is the heading angle with respect to the tangential angle of the centerline. We assume that the road width is given, which is a constraint of a lateral offset. Also, as opposed to the prior approach, which assumes for simplicity that the constraint of the initial lateral acceleration on the boundary condition is zero [12], we derive the initial lateral acceleration with reference to the arc-length, ($d^2q(0)/ds^2$) by using the current yaw rate, (ω), the tangential velocity, (v_T), and the curvature of the centerline, (k_c). Initial offset, (q_i) and curvature of the centerline, (k_c) are calculated by using the parabolic model, ($y=c_2 x^2+c_1 x+c_0$) for the centerline as presented in (3).

$$k_c = \frac{d^2y/dx^2}{\left[1 + (dy/dx)^2\right]^{3/2}} \qquad q_i = \sqrt{(x_h - x^*)^2 + (y_h - y^*)^2} \qquad (3)$$

where, (x_h, y_h) and (x^*, y^*) is the current position of the ego-vehicle and the nearest point on the centerline with the vehicle, respectively. We assume that the coefficients of the parabolic model are provided by the sensors such as camera and/or global positioning system with road information.

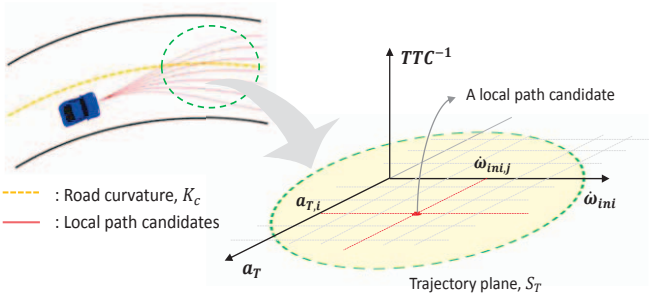

Figure 2. Trajectory plane which is compsed of the tangential acceleration and initial yaw acceleration of all local path candidates

B. Trajectory plane

In order to provide intuitive threat assessment results, the inverse TTC value are plotted on a 2-dimensional trajectory plane, (S_T) where each set of the tangential acceleration, ($a_{T,i}$) and the initial yaw acceleration, ($\dot{\omega}_{ini,j}$) values represents on the each local path candidate as Fig 2 and (4) shows.

$$S_T = (a_{T,i}, \dot{\omega}_{ini,j}) \quad for\ i \in 1,2,...N_{acc} \quad and\ for\ j \in 1,2,...N_{path} \tag{4}$$

Each local path candidate of the ego-vehicle can be projected as a single point onto the trajectory plane because each of these trajectories starts with a unique initial yaw acceleration values for any given tangential acceleration. A set of tangential acceleration values is determined based on the maximum deceleration and maximum acceleration, which is limited by the vehicle performance and the surface condition. A set of initial yaw acceleration is obtained from the differential equation of the heading angle as shown below.

$$\frac{dx_i}{dt} = v_i \cos\theta_i \quad \frac{dy_i}{dt} = v_i \sin\theta_i \quad \frac{d\theta_i}{dt} = v_i k_i \tag{5}$$

where x_i, y_i, and θ_i indicate the current vehicle's positon and the heading angle. The curvature, (k_i) and velocity, (v_i) of each local path candidate can be derived from the relationship between Cartesian coordinates and curvilinear coordinates [13], and then these are computed by substituting both the quintic polynomial with respect to the arc-length, ($q(s)$) and the derivative of this function, (dq/ds) and (d^2q/ds^2).

$$v_i = SQ\frac{ds_i}{dt} \qquad k_i = \frac{S}{Q}\left(k_c + \frac{(1-q_i k_c)(d^2 q_i/ds^2) + k_c (dq_i/ds)^2}{Q^2}\right) \tag{6}$$

$$\text{where } S = \text{sgn}(1-q_i k_c), \quad Q = \sqrt{\left(\frac{dq_i}{ds}\right)^2 + (1-q_i k_c)^2}$$

III. Trajectory Prediction of a Surrounding Vehicle

A. Target lane detection

Kinematic motion models such as the CV or CYRA model is commonly applied to predict the future behavior of a vehicle because the motion of a vehicle can be described by mathematical functions. However, the predictability of these models rapidly deteriorates, and the predicted trajectory starts to deviate from the true trajectory after one second because the future behavior of the vehicle is assumed constant [4][5]. Therefore, time-based risk indicators such as TTC computed based on these predicted trajectories are not reliable for time horizon farther than one second. To overcome this problem, we introduce the trajectory prediction of a surrounding vehicle via target lane detection. Conceptually, target lane detection can be estimated by evaluating the analogy between the current motion of a vehicle and the centerline of the driving lane [14]. To derive this mathematically, the state vector is introduced as shown in (7) [14][15].

$$X = [d_l \quad d_r \quad \beta \quad k]^T \tag{7}$$

where d_l and d_r indicate the distance from the vehicle (or the centerline) to the left and right lane boundary respectively, and β and k are the current heading angle and the curvature at the vehicle's position respectively. Using this vector, we can compute the normalized quadratic error function, (D^2) as (8).

$$D^2 = (X_L - X_H)^T \cdot (P_L + P_H)^{-1} \cdot (X_L - X_H) \tag{8}$$

where, X_L and X_H are the state vectors of the centerline and the vehicle for target lane detection, and P_L, and P_H are four-by-four covariance matrices corresponding to each state vector. We can assign the relative weights to each element of the state vector by using the covariance matrix. As a result, we can quantitatively express the discrepancy between the current vehicle motion and the centerline. If the D value is lower or equal to a threshold value, (D_{th}), the current driving lane becomes the target lane. If the D value is larger than the threshold, then the vehicle is predicted to move towards the left or right neighboring lane depending on the location of the vehicle (Section III.B). The threshold is set to two because the cumulative probability of a chi-square distribution with four degrees of freedom is equal to 0.6 at this value [16].

Furthermore, when a final time, (t_f) is fixed at a certain value, the lane change trajectories predicted for large values of D tend to show overshoot and/or oscillation due to the nature of a quintic function. This is quite unrealistic because a vehicle would try to complete the lane change motion rapidly without the overshoot as shown in Fig 3. To solve this problem, we introduce another quintic function for the final prediction time, (t_f) as a function of the D^2 value, ($f(D^2)$), as depicted in Fig 4. Using this function, the prediction time horizon is adjusted depending on the D value, so that the

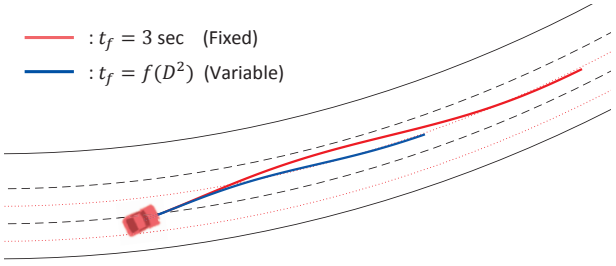

Figure 3. Trajectory prediction of a surrounding vehicle with fixed and variable final time, t_f

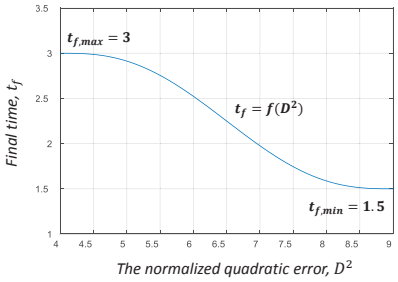

Figure 4. Final time profile with respect to the normalized quadratic error function, D^2 in case of the lane change

overshoot or oscillation is minimized. This method is computationally more efficient than the existing approach that selects the best trajectory among the candidate set generated corresponding to a different final time [14]. For lane keeping, we set the final time, t_f =1.5 *sec*, since it is completed quickly.

B. Trajectory prediction

After target lane detection is completed, future trajectory of a surrounding vehicle is predicted by using a quintic polynomial, which is equivalent to that of the ego-vehicle. The distinction between two vehicles in terms of the trajectory generation is the final position, (q_f) constraint in the boundary condition. When generating all achievable local path candidates for the ego-vehicle, a set of predetermined lateral offset from the centerline is utilized as the constraint on the final position. In contrast, the final lateral offset of the surrounding vehicle is obtained from the target lane detection as in (9).

$$q(s_f) = q_f^* = \begin{cases} L & (D \geq D_{th} \text{ and } d_{l,H_s} < d_{l,L}) \\ 0 & (D < D_{th}) \\ -L & (D \geq D_{th} \text{ and } d_{r,H_s} < d_{r,L}) \end{cases} \quad (9)$$

where, L is the lane width. Since we cannot control the surrounding vehicle in person, the motion of this vehicle involves a relatively large uncertainty than that of the ego-vehicle. Thus, the degrees of freedom in the equation of motion should be increased to cover this uncertainty. Basically, the maximum degrees of freedom in the case of polynomial is five so that the coefficients can be determined by using a given position, velocity, and acceleration. However, since the arc-length at the final prediction time, ($s(t_f)$) is a dependent variable according to the D value, the constraint imposed on the final position becomes removed. In other words, tangential motion of the surrounding vehicle is modeled by using a quartic (4th order) polynomial. Boundary conditions to determine the coefficients of this polynomial are provided below in (10).

$$s(0) = 0 \quad \frac{ds(0)}{dt} = v_{T,ini} \quad \frac{d^2s(0)}{dt^2} = a_{T,ini} \quad \frac{ds(t_f)}{dt} = v_{T,ini} \quad \frac{d^2s(t_f)}{dt^2} = 0 \quad (10)$$

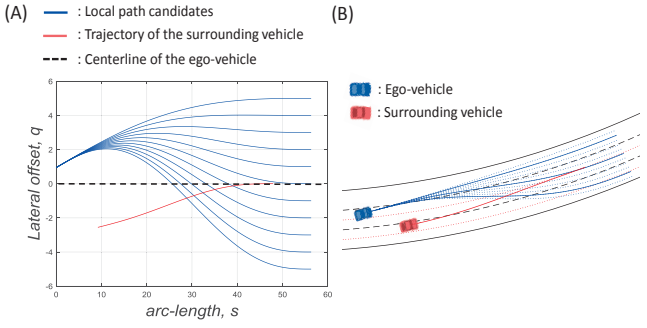

Figure 5. Example of the coordinate synchronization: (A) Relative position in curvilinear coordinates and (B) in Cartesian coordinates

IV. Threat Assessment Based on Time-to-Collision

A. Coordinate synchronization

In section IV, trajectory-based TTC computation method is presented by comparing two vehicle's remaining time to arrive at intersecting points. However, before that, the coordinates of the two vehicles must be synchronized in order to identify the intersecting (potential collision) points. To accomplish this, the centerline of the ego-vehicle is designated as the reference, and the trajectory of a surrounding vehicle is described with respect to this reference. That is, the relative position along the curvilinear coordinates, (Δs_0, Δq_0) is yielded as follows (11).

$$\Delta s_0 = \frac{1}{k_s}\theta_s \qquad \Delta q_0 = \begin{cases} \alpha \cdot L + q_{i,s} & \text{(Case I)} \\ 0 & \text{(Case II)} \\ -\alpha \cdot L + q_{i,s} & \text{(Case III)} \end{cases} \quad (11)$$

$$\text{where } \theta_s = \tan^{-1}(2a_{2,s} \cdot x_s^* + a_{1,s}) \text{ and } k_s = \frac{d^2y_s/dx^2}{[1+(dy_s/dx)^2]^{3/2}}$$

where, θ_s and k_s denote the heading angle and the curvature at the closest point on the surrounding vehicle's centerline, (x_s^*, y_s^*). Also, $q_{i,s}$ is the surrounding vehicle's initial lateral offset from its centerline. Case I, II, and, III respectively indicate the relative position of the surrounding vehicle on the left, center, and right side of the reference centerline. Also L is the lane width and α is a binary index, which is determined by whether a surrounding vehicle is driving in the same lane of the ego-vehicle. An example of coordinate synchronization is shown in Fig 5, which depicts a situation where the surrounding vehicle is predicted to initiate a left lane change.

B. Time-to-Collision (TTC)

Once the coordinates are synchronized, TTC is computed by comparing the entrance and exit time of two vehicles into and out of the conflict area in curvilinear coordinates. The conflict area means the region where the occupied spaces of two vehicles overlap. This area can be obtained by assigning lateral offsets to left and right sides of each trajectory with a half of the vehicle width as Fig.6 (A) shows. From this, four vertices of the potential collision area, ($s_{c,i}$ for i=A,B,C,D), can be identified by solving (12).

$$q_{h,j}(s) - [q_{s,j}(s - \Delta s_0) + \Delta q_0] = 0 \quad \text{for } j \in [\text{left, right}] \quad (12)$$

where, the subscripts h and s indicate the ego-vehicle and the surrounding vehicle respectively, and the left or right denoted as the subscript j is the direction of the trajectory offset with respect to the longitudinal axis of the vehicle. Note that a quintic polynomial function is employed to describe the lateral offset, ($q(s)$), and thus an analytic solution to find roots cannot be derived. Hence, a numerical method using an eigenvalue is introduced. In order to minimize the computational efforts, the numerical method is applied only to pairs of trajectories that intersect each other. The existence of roots within the range of the predicted arc-length horizon is validated through the Intermediate Value Theorem.

After finding the potential collision area, each vehicle's arrival and departure time at these locations are computed. Here, note that one of the two front or the two rear vertices of the vehicle arrives or departs from the conflict area, respectively. The entrance and the exit points are denoted as B and D in Fig. 6(A). The arrival and the departure time of each vehicle at the entrance and exit point are obtained by computing (13).

$$
\begin{aligned}
s_h(t) &= s_{entrance} - \sqrt{\left(l_{f,h}\right)^2 + \left(t_{w,h}/2\right)^2} \cdot \cos(\pi/4 - \theta_1) \\
s_h(t) &= s_{exit} + \sqrt{\left(l_{r,h}\right)^2 + \left(t_{w,h}/2\right)^2} \cdot \cos(\pi/4 - \theta_2) \\
s_s(t) &= s_{entrance} - \sqrt{\left(l_{f,s}\right)^2 + \left(t_{w,s}/2\right)^2} \cdot \sin(\pi/4 - \theta_3) \\
s_s(t) &= s_{exit} + \sqrt{\left(l_{r,s}\right)^2 + \left(t_{w,s}/2\right)^2} \cdot \sin(\pi/4 - \theta_4)
\end{aligned} \tag{13}
$$

$$
\text{where } \tan\theta_1 = \frac{dq_{h,right}(s_{entrance})}{ds},\ \tan\theta_2 = \frac{dq_{h,left}(s_{exit})}{ds},
$$

$$
\tan\theta_3 = \frac{dq_{s,left}(s_{entrance})}{ds},\ \tan\theta_4 = \frac{dq_{s,right}(s_{exit})}{ds}
$$

where, l_f and l_r are the distance from the center of gravity to the front and rear of a vehicle respectively, and t_w is the vehicle width. Since the tangential motions of the ego-vehicle and the surrounding vehicle, ($s_h(t)$ and $s_s(t)$) are modeled as a quadratic and a quartic polynomial respectively (Section II.A and III,B), an analytic formula can be used to find the roots.

After computing the arrival time of each vertex at the potential collision area, the following step is to check whether two vehicles have an overlapping time window. If the overlap time window, (T_{col}), exists as shown in Fig 6 (B), a collision is expected to occur because the partial volumes of two vehicles

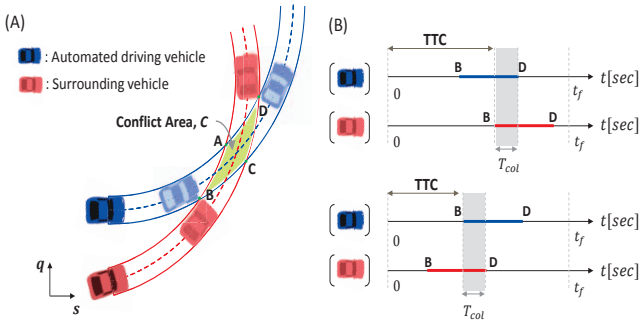

Figure.6 The trajectory-based TTC computation method: comparing the entrance and exit times at the potential collision area

coincide in the conflict area. In this case, the minimum time of the overlap time window is denoted as the TTC value. In contrast, if an overlap time interval does not exist, a collision will not occur because one vehicle exits the conflict area before the other vehicle arrives. Owing to this approach, the proposed method to compute TTC can foresee the non-conflict conditions even when two vehicle trajectories intersect with each other. The ego-vehicle is, thus, allowed to navigate more aggressively than when employing binary collision prediction, which merely detects whether each pair of trajectories intersect. Furthermore, by considering the vehicle dimensions, driving risks can be accurately predicted compared with treating the vehicle as a point mass.

V. Simulation Results

In order to provide driver assistance systems (DAS) or automated driving systems with risks associated with each local path candidate, TTC computed in the Section IV is inversed, and then projected onto the trajectory plane (Section II.B). Since the Inverse TTC (TTC^{-1}) increases with higher risks (or shorter TTC), a local path candidate with a higher value of TTC^{-1} should be avoided. To verify the effectiveness of the proposed threat prediction algorithm, the algorithm is simulated for a few scenarios over a curved highway. Simulation parameters are defined in Table I. Here, the vehicle dimensions of both the ego-vehicle and the surrounding vehicle are identical, and the vehicle is assumed to follow the target lane after completing the transient motion. The maximum tangential deceleration is -8m/s^2 obtained from a calculation using a simple equation, $a_{max}=\mu g$ based on the assumption that the road surface is dry asphalt (μ=0.8).

Fig 7 (A) and (B) show the TTC^{-1} map when the target lane of the surrounding vehicle is respectively estimated towards the left and right with reference to the current driving lane. Here, horizontal axis and vertical axis is the tangential acceleration and initial yaw acceleration respectively. Additionally, note that local path candidates that have larger initial yaw acceleration than that of the middle of set lie on above it in the TTC^{-1} map. Conversely, if a local path candidate that have smaller initial yaw acceleration lie on below the middle of set. In the simulation, the initial tangential velocity and the acceleration are set to $v_{T,h}(0)=25m/s$, $a_{T,h}(0)=0m/s^2$, $v_{T,s}(0)$

TABLE I. Simulation Parameters

Symbol	Parameter	Value
l_f	Distance from center of gravity to front (m)	1.1
l_r	Distance from center of gravity to rear (m)	1.24
t_w	Width of vehicle (m)	1.78
L	Width of lane (m)	4
$a_{T,i}$	Tangential acceleration set (m/s^2)	[-8,2]
D_{th}	Threshold for target lane detection	2
$t_{f,h}$	Final prediction time of the automated driving vehicle	2.5
$t_{f,s}$	Final prediction time of the surrounding vehicle	[1.5, 3]

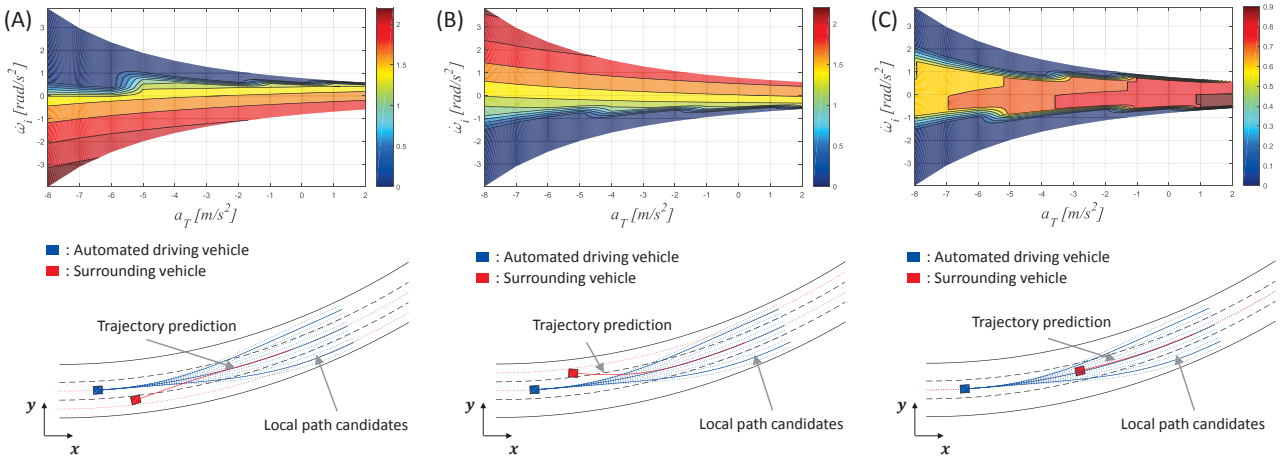

Figure.7 Inverse TTC (TTC^{-1}) map corresponding to the trajectory prediction of the surrounding vehicle

$=20m/s$, $a_{T,s}(0)=2m/s^2$. In Fig 7(A) and (B), if the surrounding vehicle is driving in close proximity to the ego-vehicle, there still exists collision risks even though maximum deceleration is initiated. Therefore, to safely avoid a collision in this case, both decelerating and steering should be applied simultaneously. If the distance between two vehicles becomes greater than that in the prior situation, the ego-vehicle can employ a variety of control strategies for collision avoidance. This is because the arrival time of two vehicles at the potential collision locations is different despite that their trajectories intersect. Fig 7 (C) shows the TTC^{-1} map when two vehicles are driving on the same lane, and similarly collision risks cannot be eliminated when only applying maximum deceleration. Therefore, a combination of both steering and deceleration is required to prevent a collision at high speed.

VI. Conclusion

In this paper, we propose a threat prediction algorithm that can accurately predict TTC over a longer time based on future trajectory predictions of the surrounding vehicle. Consequently, a long-term risk analysis can be performed by computing the TTC value higher than 1 sec, and thus, the vehicle is allowed to have sufficient time to respond to any hazard. Also, as plotting collision risks denoted by TTC^{-1} on the trajectory plane, a control strategy for collision avoidance can be determined approximately. However, since the future can never be perfectly predicted, the accuracy of the proposed algorithm may degrade for far future predictions. Thus, in future work, we will associate the uncertainty of the future motion with the trajectory prediction from a stochastic perspective. We will also focus on a methodology to choose a desirable control strategy based on the assessed threat.

References

[1] M. Brännström, J. Sjöberg, and E. Coelingh, "A Situation and Threat Assessment Algorithm for a Rear-End Collision Avoidance System," in *Proc. IEEE Intell. Vehicles Symp.*, Eindhoven, 2008, pp.102-107.

[2] F. Wang, M. Yang, and R. Yang, "Conflict-Probability-Estimation -Based Overtaking for Intelligent Vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 2, pp. 366-370, Jun., 2009.

[3] J. Ward, G. Agamennoni, S. Worrall, and E. Nebot, "Vehicle Collision Probability Calculation for General Traffic Scenarios Under Uncertainty," *in Proc. IEEE Intell. Vehicles Symp.*, Michigan, 2014, pp. 986-992.

[4] W. Yaw, H. Xhao, P. Bonnifait, and H. Zha, "Lane Change Trajectory Prediction by using Recorded Human Driving Data," *in Proc. IEEE Intell. Vehicles Symp.,* Gold Coast, 2013, pp. 430-436.

[5] S. Lefèvre, D. Vasquez, and C. Laugier, "A Survey on Motion Prediction and Risk Assessment for Intelligent Vehicles," *Robomech Journal*, vol. 1, no. 1, pp. 1-14, 2014.

[6] D. Ferguson, M. Darms, C. Urmson, and S. Kolski, "Detection, Prediction, and Avoidance of Dynamic Obstacles in Urban Environments," *in Proc. IEEE Intell. Vehicles Symp.*, Eindhoven, 2008, pp. 1149-1154.

[7] A. Lawitzky, D. Wollherr, and M. Buss, "Maneuver-based Risk Assessment for High-Speed Automotive Scenarios," *in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura, 2012, pp. 1186-1191.

[8] D. Ferguson, T. M. Howard, and M. Likhachev, "Motion Planning in Urban environments," *in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nice, 2008, pp. 1070-1076.

[9] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, "Optimal Trajectory Generation for Dynamic Street Scenarios in a Fren´et Frame," *in Proc. IEEE International Conference on Robotics and Automation*, Anchorage, 2010, pp. 987-993.

[10] K. Chu, M. Lee, and M. Sunwoo "Local Path Planning for Off-Road Autonomous Driving With Avoidance of Static Obstacles," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 4, pp. 1599-1616, Dec., 2012.

[11] M. Knoop, and T. Haeussler, "Method for Determining an Evasion Trajectory for a Motor Vehicle, and Safety Device or Safety System," U.S. Patent 0067252 A1, Mar. 6, 2014.

[12] S. Ammoun, F. Nashashibi, and C. Laurgeau, "An Analysis of the Lane Changing Manoeuvre on Roads: the Contribution of Inter-Vehicle Cooperation via Communication," *in Proc. IEEE Intell. Vehicles Symp.*, Istanbul, 2007, pp. 1095-1100.

[13] T. D. Barfoot, and C. M. Clark, "Motion Planning for Formations of Mobile Robots," *Robotics and Autonomous Systems*, vol. 46, no. 2, pp. 65-78, 2004.

[14] A. Houenou, P. Bonnifait, V. Cherfaoui, W. Yao, "Vehicle Trajectory Prediction based on Motion Model and Maneuver Recognition," *in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, Japan, 2013.

[15] D. Muller, J. Pauli, M. Meuter, L. Ghosh, and S. M.-Schneiders, "A Generic Video and Radar Data Fusion System for Improved Target Selection," *in Proc. IEEE Intell. Vehicles Symp.*, Baden-Baden, 2011, pp. 679-684.

[16] A. Hayter, *Probability and Statistics for Engineers and Scientists 4th edition.* Cengage Learning, 2012.