

Occlusion-Aware Risk Assessment for Autonomous Driving in Urban Environments

Ming-Yuan Yu¹, Ram Vasudevan², and Matthew Johnson-Roberson³

Abstract—Navigating safely in urban environments remains a challenging problem for autonomous vehicles. Occlusion and limited sensor range can pose significant challenges to safely navigate among pedestrians and other vehicles in the environment. Enabling vehicles to quantify the risk posed by unseen regions allows them to anticipate future possibilities, resulting in increased safety and ride comfort. This paper proposes an algorithm that takes advantage of the known road layouts to forecast, quantify, and aggregate risk associated with occlusions and limited sensor range. This allows us to make predictions of risk induced by unobserved vehicles even in heavily occluded urban environments. The risk can then be used either by a low-level planning algorithm to generate better trajectories, or by a high-level one to plan a better route. The proposed algorithm is evaluated on intersection layouts from real-world map data with up to five other vehicles in the scene, and verified to reduce collision rates by 4.8× comparing to a baseline method while improving driving comfort.

Index Terms—Collision Avoidance, Motion and Path Planning, Simulation and Animation.

I. INTRODUCTION

ADVANCEMENTS in sensing technology and algorithmic improvements bring the reality of everyday autonomous driving closer to fruition. Light Detection and Ranging (LiDAR) sensors enable the construction of 3D maps [1] and can see tens or hundreds of meters away, even at night [2]. High definition cameras capture images that can be used for tasks such as semantic segmentation [3] and object detection [4]. Many tasks can be performed at levels surpassing that of humans thanks to recent developments in deep neural networks [5].

However, all sensors still have limited sensing capabilities. LiDARs and cameras, for instance, have difficulty identifying objects beyond a certain distance due to finite range, sensitivity, and angular resolution. In addition, both of these sensors can not see through opaque objects which could result in large unobserved regions. An illustration of such a scenario is shown in Fig. 1.

One of the reasons why human drivers can safely navigate even under occlusions is that they augment their sensing capa-

This work was supported by a grant from Ford Motor Company via the Ford-UM Alliance under Award N022884. (Corresponding author: Ming-Yuan Yu.)

¹M.-Y Yu is with Robotics Institute, University of Michigan, Ann Arbor, MI 48109, USA myyu@umich.edu

²R. Vasudevan is with the Department of Mechanical Engineering, the University of Michigan, Ann Arbor, MI 48109, USA ramv@umich.edu

³M. Johnson-Roberson is with the Department of Naval Architecture and Marine Engineering at the University of Michigan, Ann Arbor, MI 48109, USA mattjr@umich.edu

Digital Object Identifier (DOI): 10.1109/LRA.2019.2900453.

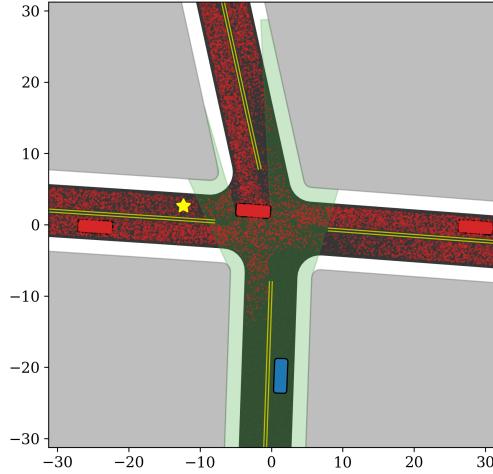


Fig. 1: Ego vehicle (blue box) intends to perform an unprotected left turn to the goal (yellow star) at an intersection. The irregular shape of the observable polygon (green shaded region) is caused by 1) limited sensor range and 2) occlusions from other vehicles (red boxes) and buildings (gray regions.) Our algorithm quantifies the distribution of risk (red particles) posed by other vehicles including the ones which are outside of the observable polygon. This is possible under the assumption that we know the geometry of road layout and the nominal (or worst case) speed of other vehicles at this particular intersection. Both axes are in meters.

bilities by leveraging semantic and geometrical information of the environment, and anticipate the need to slow down due to the potential risk of collision that arises due to occlusions [6], [7]. In addition, earlier braking would reduce the maximum deceleration which consequently leads to greater ride comfort.

To provide an example from the real-world imagine the following: pulling up to a left turn next to a tall tree or building, similar to the scenario shown in Fig. 1. Typically a driver leans forward and pulls the car slightly ahead to see into oncoming traffic before completing the turn. In the driver's mind, they have a map of the unseen spaces and know that a car could emerge from beyond the current line of sight. As a result, they proceed cautiously to try to improve their visibility and do not turn until they can confirm a sufficient gap in the traffic.

This paper presents an algorithm that encodes this form of human driving by quantifying the risk caused by limited sensing capabilities and geometric occlusion. The proposed algorithm can be used to make autonomous vehicles navigate

safely with improved ride comfort in urban environments, and it is agnostic to how the vehicle makes decisions.

The paper is organized as follows: Section II reviews related work in the field of risk assessment and planning under occlusion. Section III describes how our algorithm leverages the known road layout to quantify risk in the environment, and demonstrates how it can be easily integrated with a simple planning algorithm. Section IV introduces two baseline methods and our evaluation methodologies. Section V evaluates the proposed occlusion-aware method, and shows that statistically our algorithm performs significantly better in terms of collision rate and ride comfort on both synthetic and real-world intersections. Section VI concludes and discusses future directions of this work.

II. RELATED WORK

Most of the previous work on motion prediction and risk assessment can be categorized into one of two following categories. The first category quantifies the risk as the probability of having a collision with another vehicle or pedestrian. The second category assesses risk as the degree of deviation from a nominal set of behaviors (e.g. veering from the lane rapidly). A well-organized survey is given in [8]. This paper addresses the first category of problems with a specific focus on collisions caused by occluded objects.

Prior work has addressed the issue of occlusion from a tracking perspective. Wyffels and Campbell [9] keep track of obstacles in occluded areas by utilizing negative information under the assumption that undetected objects are not likely to appear in visible space. Yu and LaValle [10] maintain the tracks of targets that move outside the field-of-view by formulating the problem as a pursuit evasion game. Galceran, Olson, and Eustice [11] augment states of a standard tracker to estimate occluded states for other agents and provide more robust data association when the occluded agents reappear in the scene. Ondruska and Posner [12] and Dequaire, Ondrka, Rao, *et al.* [13] utilize recurrent neural networks trained in an unsupervised manner and are able to track occluded object from only raw, occluded sensor data. Although these models keep tracks of missing targets that enter occluded regions, they all need at least one detection to start tracking. They do not explicitly handle risks caused by potential incoming traffic which is occluded or outside the sensor horizon and thus never detected in the first place.

Partially Observable Markov Decision Process (POMDP) is a common approach to tackle decision making problems under uncertainty and consequently can implicitly handle probabilistic occlusion. Brechtel, Gindel, and Dillmann [14] use Monte Carlo Value Iteration and Brechtel, Gindel, and Dillmann [15] show it is possible to optimize a continuous POMDP model. Bouton, Nakhaei, Fujimura, *et al.* [16] approximate the global solution by solving a POMDP for each agent independently through utility fusion [17]. The reduction in state space required to make these approaches tractable limits their applicability, particularly for real-time high speed driving. The algorithm we present in this paper differs in both goal and implementation. We focus on quantifying risk in the

environment instead of the risk associated with the actions of the ego vehicle. Our algorithm is agnostic to planning and so could be coupled with a POMDP or any other planning or decision making algorithms.

Most closely related to the approach presented here are two risk quantification approaches [6], [7]. Orzechowski, Meyer, and Lauer [6] over-approximate all possible states of the incoming traffic by considering the leading edges of the visible polygon. Although safety is guaranteed, the resulting over-approximated polygons are not probabilistic, whereas our approach captures the full distribution of risk. Lee, Jo, and Sunwoo [7] perform probabilistic risk assessment by utilizing prebuilt high definition maps. While the results in [7] look promising, they do not show how their risk assessment could be used for planning to achieve safer driving. Furthermore, both [6] and [7] show very limited results with only a single additional vehicle in the scene and it is unclear how these approaches perform in crowded scenes such as urban intersections. We focus on realistic intersections derived from real map data and occupied with many vehicles.

Our approach presents several novel contributions: 1) we present an algorithm which performs probabilistic risk assessment of both observed and unobserved regions at urban intersections; 2) the approach is control algorithm agnostic and can be integrated with any deterministic or probabilistic planning approach; 3) we derive risk assessment from large-scale map data and extensively evaluate our approach with up to five other vehicles in the scene, and show significant reduction in collision rate and increase in ride comfort.

III. METHOD

We first describe our method in probabilistic risk assessment in Section III-A. We generate a distribution over the Cartesian space. In Section III-B we show how to integrate the risk to a simple optimization-based planning algorithm and describe the primary cost function.

A. Risk Assessment Over Cartesian Space

High Definition (HD) maps are used commonly in autonomous driving [7]. These maps have rich data about intersections and can encode information such as nominal trajectories and maximum speed of all traffic through a region. Assuming that the map and the location of the ego vehicle are known, an *observable polygon* can be generated for a vehicle's sensor configuration (maximum range, angular resolution, and field of view) without the actual sensor returns. Here we focus on LIDARs, but the principles remain the same for other sensor modalities. The shape of the observable polygon is constrained by occlusions caused by objects such as other vehicles, trees, and buildings. With the observable polygon, one can identify free space at the current time. However, the current observable polygon alone can provide little information about long-term risk.

Current free space estimates are insufficient for planning for the future as vehicles can suddenly appear from regions outside of the observable polygon. In order to quantify the risk due to limited sensing in the context of long-term planning,

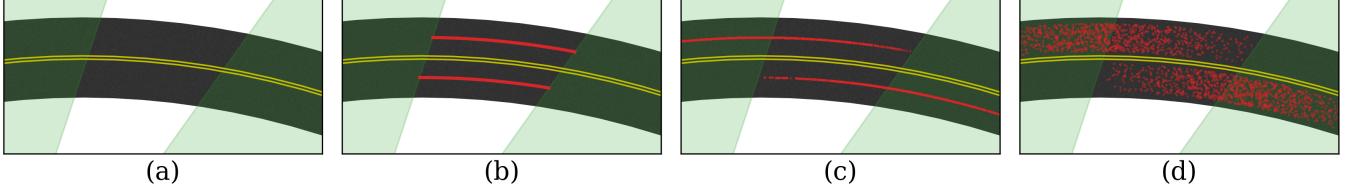


Fig. 2: Illustration of our algorithm on a (a) partially observed road, where the green shaded regions are within the sensor's field-of-view. (b) Firstly, assuming that the map and the ego vehicle's location are known, the centerlines of the unobserved road segments (red) are extracted. (c) Secondly, we sample particles $\{(s^{[i]}, v^{[i]})\}_{i=1}^{N_k}$ along the extracted splines $c_k, k \in \{1, 2\}$, where the location s and speed v are drawn from uniform distributions, and propagate each particle forward in time assuming constant speed. (d) Finally, a random offset perpendicular to each centerline is added to each particle to incorporate the non-zero size of potentially unobserved vehicles.

we need to consider vehicles that are potentially hidden in unobserved regions. We leverage the paradigm of the particle filter to perform this prediction. Particles are used to represent the distribution of potential vehicle locations originated from unobserved regions. This approach was selected because of its simplicity and parallelizability.

Consider a scenario with two lanes shown in Fig. 2. We represent the lanes of travel by cubic splines. Each cubic spline c_k is parameterized by its position s along the spline:

$$c_k(s) = \begin{bmatrix} x_k(s) \\ y_k(s) \end{bmatrix}, s \in [0, \bar{s}_k]$$

$$k \in \{1, 2, \dots, M\}$$

where $[x_k(s) \ y_k(s)]^\top$ is the position of a point on c_k at s , M is the number of lanes in the scene, and \bar{s}_k is the total length of the spline c_k .

We first extract all possible centroids for all valid vehicle positions in all lanes of travel in the unobserved regions. On each spline c_k , we consider L_k disjoint unobserved segments. A set of N_k particles $\{(s^{[i]}, v^{[i]})\}_{i=1}^{N_k}$ are sampled independently from uniform distributions in these unobserved segments, where $s^{[i]}$ and $v^{[i]}$ are the position and speed of the i -th particle. The position s and speed v is distributed as follows:

$$s \sim U \left(\bigcup_{j=1}^{L_k} [\underline{s}_j, \bar{s}_j] \right), \quad v \sim U([\underline{v}_k, \bar{v}_k])$$

where $[\cdot, \cdot]$ is a closed set between two real numbers, $U(\cdot)$ is an uniform distribution on a set, \underline{s}_j and \bar{s}_j is the starting and ending position of an unobserved segment j on spline c_k , \underline{v}_k and \bar{v}_k are the minimum and maximum speed of other vehicles, respectively. Uniform sampling is used because we assume no prior knowledge of s and v . A more descriptive distribution can be used if some prior information is available.

Assuming that each particle is traveling with a constant speed, we can then propagate all the particles forward in time for T_f seconds:

$$\hat{s}^{[i]} = s^{[i]} + v^{[i]} \cdot T_f$$

where $\hat{s}^{[i]}$ is the position of the i -th particle after T_f seconds. This results in a distribution of particles along the centerline

of each lane, as shown in Fig. 2c. Note that more sophisticated motion models with variable speed can also be used at a cost of more computation time.

To account for the size of vehicles and lateral displacements within the lane, an offset $b^{[i]}$ is sampled from an uniform distribution $U([-\bar{b}, \bar{b}])$ and added to each particle in Cartesian space perpendicular to the spline. The uniform distribution is chosen since we assume no prior information for the position of the occluded vehicles in the lateral direction.

$$u_k(s) := \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \cdot \frac{\partial c_k}{\partial s}(s)$$

$$\hat{p}_k^{[i]} = c_k \left(\hat{s}^{[i]} \right) + \frac{b^{[i]}}{\|u_k(\hat{s}^{[i]})\|_2} \cdot u_k \left(\hat{s}^{[i]} \right)$$

where $\|\cdot\|_2$ is the 2-norm of a vector, u_k is the unnormalized vector perpendicular to c_k , and \bar{b} is the maximum deviation among all the particles from their corresponding centerline. We define the set $\{\hat{p}_k^{[i]}\}_{i=1}^{N_k}$ to be the distribution of risk over the Cartesian space on lane k after T_f seconds, as shown in Fig. 2d.

For observed vehicles, we model them as rectangles along valid lanes. To incorporate them into our proposed formulation, we treat them similarly. For each vehicle, we extract the spline segments within the corresponding rectangle, and apply the aforementioned method as if the segments are in the unobserved regions. This makes our algorithm conservative and performs reasonably well even when the intention and speed of other vehicles are unknown or noisy. If a good tracker is also available, the planner can potentially behaves more aggressively without sacrificing safety.

The overall distribution of risk $\{\hat{p}^{[i]}\}_{i=1}^N$ is simply the union of all sets.

$$\left\{ \hat{p}^{[i]} \right\}_{i=1}^N = \bigcup_{k=1}^M \left\{ \hat{p}_k^{[i]} \right\}_{i=1}^{N_k}$$

where N is the total number of particles in the scene on M lanes.

This risk over Cartesian space can be easily integrated with any control or planning algorithm as either the primary cost or in conjunction with other costs as an auxiliary cost function. In addition, it can also be used along with any existing risk

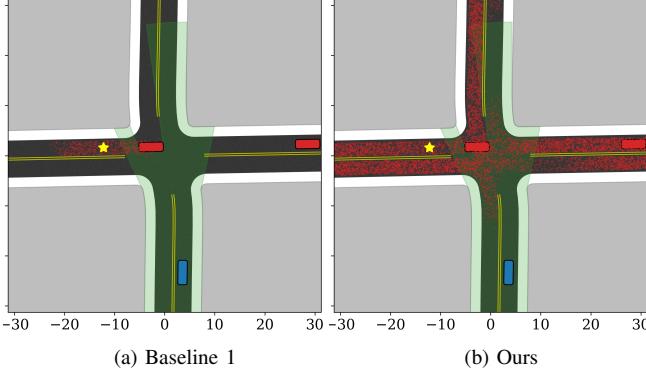


Fig. 3: Comparison between the (a) baseline and (b) proposed method. The baseline method only predicts distribution of risk (red particles) caused by observed vehicles, whereas the proposed method also predicts the risk caused by unobserved regions.

assessment method designed for only observed vehicles. In Section III-B we show how we can utilize the risk $\{\hat{p}^{[i]}\}_{i=1}^N$ as the major cost function of a simple optimization-based planning algorithm.

B. Planning

In this subsection we demonstrate how the risk described in Section III-A can be used in practice. We integrate it into a optimization-based planning algorithm and show improvements in both safety and ride comfort. The rudimentary planner used here can be replaced by any other cost-based planners as the technique is agnostic to planning approach.

Assuming that at time t the ego vehicle travels with speed v_{ego} on the intended route c_{ego} , which is also a cubic spline parameterized by its position s_{ego} along the spline. The planner first considers the *safety cost* $J_1(a_{ego})$ associated with an acceleration (or deceleration) a_{ego} :

$$J_1(a_{ego}) = \sum_{i=1}^N f^{[i]}(a_{ego})$$

where $f^{[i]}$ is the potential function of the i -th particle which is positive when the particle $\hat{p}^{[i]}$ is within the ego lane and zero otherwise. The function $f^{[i]}$ is defined as follows:

$$\begin{aligned} f^{[i]}(a_{ego}) &:= \begin{cases} \exp\left(-\frac{r^{[i]}(a_{ego})^2}{\sigma^2}\right) & , \text{if } d^{[i]} \leq \bar{b} \\ 0 & , \text{otherwise} \end{cases} \\ \hat{s}_{ego} &:= s_{ego} + v_{ego} \cdot T_f + \frac{1}{2} a_{ego} \cdot T_f^2 \\ r^{[i]}(a_{ego}) &:= \|c_{ego}(\hat{s}_{ego}) - \hat{p}^{[i]}\|_2 \\ d^{[i]} &:= \inf_s \|c_{ego}(s) - \hat{p}^{[i]}\|_2 \end{aligned}$$

where \hat{s}_{ego} is the future position of the ego vehicle along c_{ego} , $r^{[i]}(a_{ego})$ is the distance between particle $\hat{p}^{[i]}$ and $c_{ego}(\hat{s}_{ego})$, $d^{[i]}$ is the minimum distance between particle $\hat{p}^{[i]}$ and the ego vehicle's intended route c_{ego} , and σ is the bandwidth of the

TABLE I: Parameters for Simulations

Parameter	Value
Forecast horizon, T_f	1.5 s
Replan period, T_p	0.1 s
Vehicle length, l_v	4.88 m
Vehicle width, w_v	1.86 m
Number of particles, $N_k \forall k$	$\leq 2^{15}$
Weight, λ	$2^{14} \cdot 10^{-6}$
Bandwidth, σ	$0.5l_v$
Max. offset, \bar{b}	$0.75w_v$
Desired speed, v_{des}	10 m/s
Min. speed, $\underline{v}_k = \underline{v}_{ego} \forall k$	0 m/s
Max. speed, $\bar{v}_k = \bar{v}_{ego} \forall k$	12 m/s
Min. acceleration, \underline{a}_{ego}	-8 m/s ²
Max. acceleration, \bar{a}_{ego}	2.5 m/s ²
Threshold acceleration, a_{thresh}	4 m/s

repulsive potential field. In practice, particles with $r^{[i]}(a_{ego}) \geq 2\sigma$ are discarded to speed up the calculation.

In addition to the safety cost $J_1(a_{ego})$, a *speed cost* $J_2(a_{ego})$ is also considered to drive the ego vehicle to meet the desired speed v_{des} .

$$J_2(a_{ego}) = |v_{ego} + a_{ego} \cdot T_f - v_{des}|$$

where $|\cdot|$ is the absolute value of a scalar. The optimal acceleration between time t and $t+T_p$ can be found by solving the following optimization problem:

$$\begin{aligned} \min_{a_{ego}} \quad & J_1(a_{ego}) + \lambda \cdot J_2(a_{ego}) \\ \text{s.t.} \quad & \underline{v}_{ego} \leq v_{ego} + a_{ego} \cdot T_f \leq \bar{v}_{ego} \\ & \underline{a}_{ego} \leq a_{ego} \leq \bar{a}_{ego} \end{aligned}$$

where T_p is the replan time, λ is the weight affecting how aggressive the ego vehicle behaves, \underline{v}_{ego} and \bar{v}_{ego} are the minimum and maximum speed of the ego vehicle, and \underline{a}_{ego} and \bar{a}_{ego} are the maximum deceleration and maximum acceleration, respectively. Note that a smaller λ favors more conservative behaviors. We find the value of λ in Table I is a good compromise between safety and efficiency. As shown here, the proposed risk assessment method can be incorporated with any optimization-based planner.

IV. EVALUATION

To demonstrate how safety and comfort can be improved by our algorithm, we compare it to a baseline approach which only models observed vehicles at intersections. In particular, we focus on scenarios where the ego vehicle tries to make difficult maneuvers such as an unprotected left turn.

A. Simulation

We simulate various random scenarios with five other vehicles in the scene. Each vehicle travels on a random route at a constant speed ranging from 4 to 12 m/s. A valid combination of trajectories is generated by rejection sampling so that there is no collision or overlap among the simulated vehicles.

Here we focus on four-way, un-signaled intersections for compactness and not on T- or Y-junctions, but the proposed

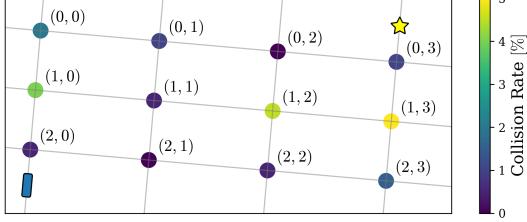


Fig. 4: Illustration of collision rates overlaid on a map with 12 intersection. A high-level planner can plan a route based on the collision rates, taking the route with the lower collision rates: $(2, 0) \rightarrow (2, 1) \rightarrow (1, 1) \rightarrow (0, 1) \rightarrow (0, 2) \rightarrow (0, 3)$.

approach conceptually generalizes. The layout of intersections can be either synthetic or from real-world map data. For the synthetic layout, the roads are constructed using straight and perpendicular segments. For real-world layouts, we obtain the geometry information from 73 real-world intersections extracted from OpenStreetMaps (OSMs) around Ann Arbor, Michigan.

To simulate scenarios with heavy occlusion, buildings are added to the map with a 2 m buffer from the boundary of the driving surface. The ego vehicle starts 15 m before the stopline with initial speed 10 m/s, and tries to perform an unprotected left turn, as shown in Fig. 3. More details of the parameters used in the simulator are listed in Table I. Note that as length of unobserved segments vary, N_k is calculated dynamically such that the density of particles stays constant at 2^{15} particles per 100 m.

B. Baseline 1

An occlusion-unaware risk assessment method is used as a baseline for comparison. The baseline method predicts distribution of risk using the exact same method described in Section III-A, but only for the regions intersecting with the observed vehicles. The same planning algorithm described in III-B is used with both the baseline and proposed method throughout all simulations.

C. Baseline 2

To the best of our knowledge, [6] is the only method that describes both risk assessment and planning in the same article. However, only a few examples were shown in their paper. Here, we recreated the scenarios in their paper and compare the speed and acceleration profiles.

We recreated the scenarios by using the same map, initial speed, desired speed, maximum acceleration and minimum acceleration provided in [6]. The initial position and the speed of the other vehicle in both scenarios are estimated by measuring the plots in [6] using AutoCAD.

D. Metrics

For baseline 1, we first simulate 2000 random scenarios with the ego vehicle performing an unprotected left turn at each intersection with the baseline method, then simulate the

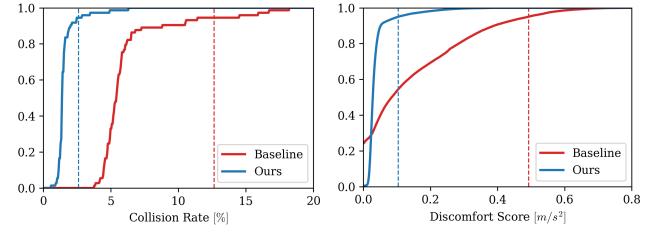


Fig. 5: CDFs of (a) collision rates and (b) discomfort scores among all 73 real-world intersections. Our method outperforms the baseline where the 95th percentile of the collision rate (vertical dashed lines) decreases from 12.64% to 2.61%, and the 95th percentile of the discomfort decreases from 0.4925 to 0.1043.

exact same set of experiments with identical trajectories with the proposed method. For each intersection, we calculate its collision rate for both methods as follows:

$$\text{Collision Rate} = \frac{\# \text{ of simulations with collision}}{\text{Total } \# \text{ of simulations}} \times 100\%$$

This meta-collision rate for a given intersection can be used by a high-level planner, which needs to plans a route between two points across a city. Overlaying the collision rates with associated intersections, a high-level planner can avoid dangerous intersections, as show in Fig. 4.

Speed and acceleration profiles are also calculated to quantify ride discomfort. However, to the best of our knowledge, there is no common computational metric in the literature for ride comfort. Typically, the literature reports thresholds on acceleration and jerk as the metric for ride comfort[18]. We define the following *discomfort score* to represent a continuous range of discomfort.

$$\text{Discomfort Score} = \frac{1}{T} \int_0^T \max(0, |a_{ego}(t)| - a_{thresh}) dt$$

where T is the duration to reach the goal and $a_{thresh} > 0$ is a threshold set to be half of the maximum deceleration.

For baseline 2, speed and acceleration profiles of two scenarios under occlusion are investigated. The first scenario illustrates an ego vehicle traveling at an intersection with no other vehicle present in the scene. The second scenario adds one vehicle coming from the left.

V. RESULTS

The distributions of both collision rate and discomfort score are discussed in this section, as shown in Fig. 5a and 5b as CDFs. In particular, the median and the 95th percentile are reported. The former describes the nominal behavior, whereas the latter represents the near-worst case.

A. Baseline 1 - Collision Rate

We first evaluate the collision rate of both the baseline and proposed method. By utilizing our algorithm, collision rates drop significantly. At the synthetic intersection, the rate

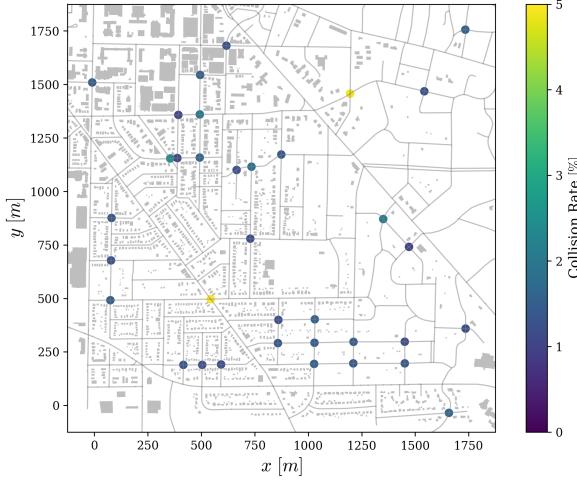


Fig. 6: Collision rates of a subset of intersections overlaid on a real-world map. A high-level planner can utilize this information to avoid dangerous intersection such as the two yellow ones.

decreases by $4.1\times$, from 5.75% to 1.40%. Results at real-world intersections also show similar improvements. Among all the 73 real-world intersections, the median collision rate decreases by $3.7\times$, from 5.40% to 1.45%, and the 95th percentiles decreases by $4.8\times$, from 12.64% to 2.61%, as shown in Fig. 5a.

The distribution of the simulated collision rates of the 73 real-world intersections is overlaid in Fig. 6. This can be added as extra information to a high-level planner to plan a route with lower collision rates. This enables a vehicle to reason about safety prior to embarking on a journey. It also enables urban planners and civil engineers to reason about safety of interactions for autonomous vehicles in a systematic and quantitative way.

B. Baseline 1 - Ride Comfort

In addition to safety, which is evaluated as collision rate in Section V-A, another benefit from our algorithm is more ride comfort. At the synthetic intersection, the median discomfort score is reduced by $2.9\times$, from 0.0795 to 0.0271. The 95th percentile of the discomfort score decreases by $10\times$, from 0.4687 to 0.0466. Similarly, the median score among all real-world intersections is reduced by $3\times$, from 0.0849 to 0.0284, and the 95th percentile of the score decreases by $4.7\times$, from 0.4925 to 0.1043.

An illustration of the synthetic and real-world intersections are shown in Fig. 8. In both cases, the baseline method has larger variations in both speed and acceleration, which means that the ego vehicle can abruptly brake only after other vehicles enter the observable polygon. On the other hand, our method naturally introduces a *virtual* stop sign, slowing the ego vehicle down even when there is no other vehicle in the observable polygon, which generates a consistent behavior across all simulations.

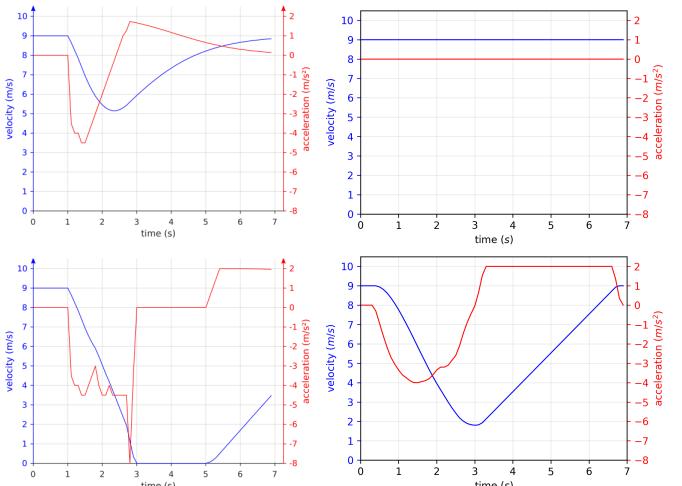


Fig. 7: Speed and acceleration profiles of the two scenarios in [6]. Top: with no other vehicle; bottom: with one other vehicle. The left column is adapted from [6].

C. Baseline 2

Both [6] and the proposed algorithm traverse through the intersection safely without collision. But our algorithm performs quantitatively better in terms of ride comfort, as shown in Fig. 7a and 7b. For the first scenario, our method obtains zero acceleration, whereas the baseline unnecessarily decelerates. The second scenario further highlights the benefit of using a probabilistic risk assessment approach, as opposed to a deterministic one proposed in [6]. The ego vehicles with our algorithm accelerate and decelerate gracefully without any jitter, while the baseline shows large jerk which leads to an uncomfortable ride experience.

VI. CONCLUSIONS

We propose a probabilistic risk assessment algorithm for autonomous driving under occlusion. We show how it can be integrated with a simple planning algorithm, and compare the proposed algorithm with a baseline method which only performs risk assessment for observed vehicles. We evaluate our algorithm in terms of collision rate and ride comfort with a large number of simulations at one synthetic and 73 real-world intersections. The results show that the proposed algorithm reduces the collision rate by up to $4.8\times$ and increase comfort by up to $10\times$. Our method shows great potential for improving both safety and comfort for autonomous driving in urban environments.

Future directions include incorporating information from other vehicles for cooperative planning, and handling heterogeneous traffic scenarios with various kinds of vehicles and pedestrians.

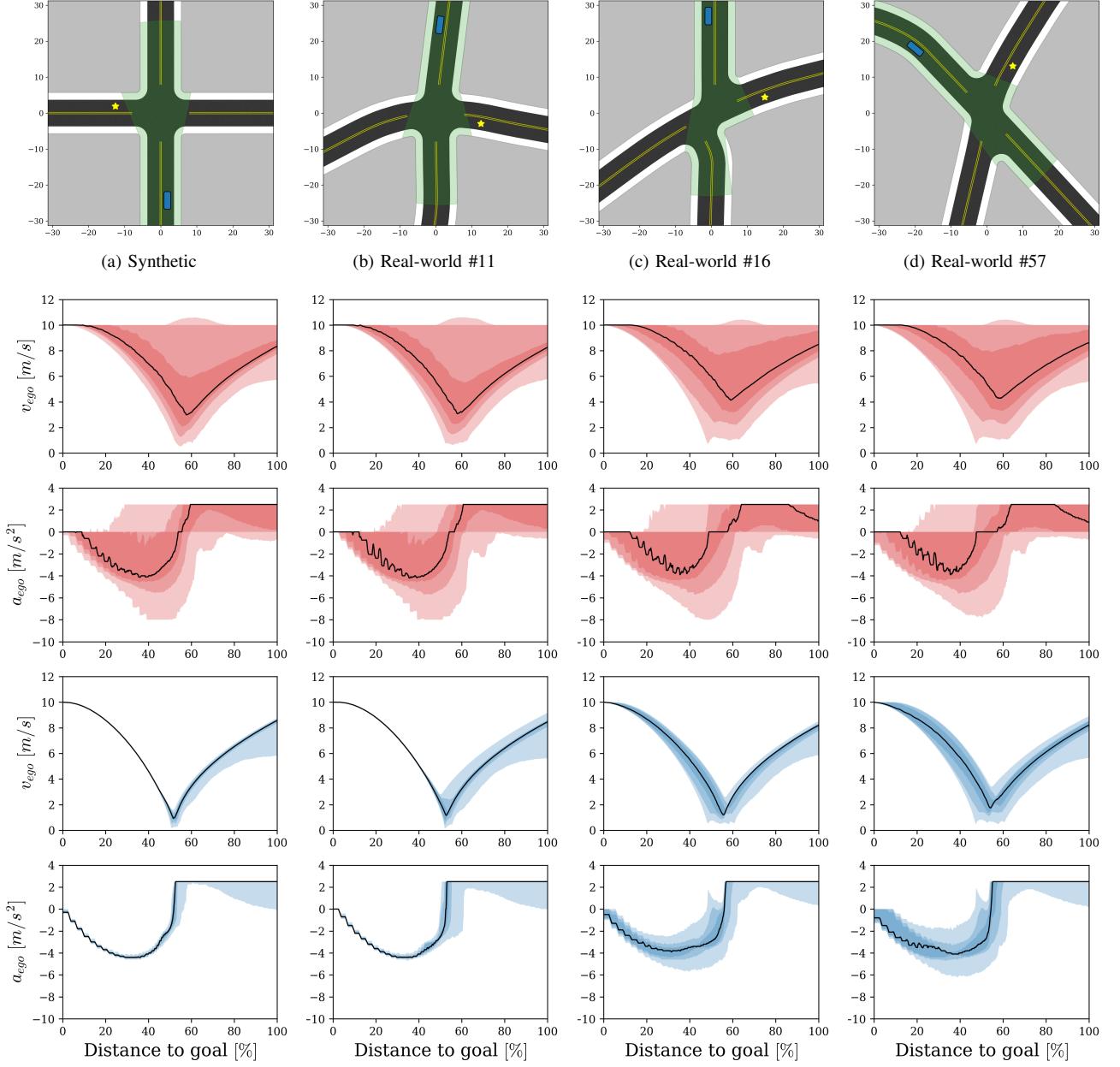


Fig. 8: Speed and acceleration profiles for baseline 1 and the proposed method at various intersections. From top to bottom: map, speed profile of baseline 1, acceleration profile of baseline 1, speed profile of our method, acceleration profile of our method. The medians of profiles are shown in solid black lines, and the percentiles are shown in different shades of colors (from dark to light: $50 \pm 15\%$, $50 \pm 30\%$, $50 \pm 45\%$.) In both synthetic and real-world intersections, the baseline method shows large variations due to abrupt braking. In addition, the deceleration can reach down to $-8 m/s^2$, which can be very uncomfortable. On the other hand, our method predominantly stays above $-4 m/s^2$ and shows smaller variations, which indicates that it performs consistently well across all simulations.

REFERENCES

- [1] S. Kammel and B. Pitzer, "Lidar-based lane marker detection and mapping," in *IEEE Intelligent Vehicles Symposium*, 2008, pp. 1137–1142.
- [2] P. Sudhakar, K. A. Sheela, and M. Satyanarayana, "Imaging lidar system for night vision and surveillance applications," in *4th International Conference on Advanced Computing and Communication Systems (ICACCS)*, 2017, pp. 1–6.
- [3] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, 2017.
- [4] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C. Fu, and A. C. Berg, "SSD: single shot multibox detector," *CoRR*, vol. abs/1512.02325, 2015.
- [5] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, Washington, DC, USA: IEEE Computer Society, 2015, pp. 1026–1034.
- [6] P. F. Orzechowski, A. Meyer, and M. Lauer, "Tackling occlusions & limited sensor range with set-based safety verification," *ArXiv preprint arXiv:1807.01262*, 2018.
- [7] M. Lee, K. Jo, and M. Sunwoo, "Collision risk assessment for possible collision vehicle in occluded area based on precise map," in *IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, 2017, pp. 1–6.
- [8] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *ROBOMECH Journal*, vol. 1, no. 1, p. 1, 2014.
- [9] K. Wyffels and M. Campbell, "Negative information for occlusion reasoning in dynamic extended multiobject tracking," *IEEE Transactions on Robotics*, vol. 31, no. 2, pp. 425–442, 2015.
- [10] J. Yu and S. M. LaValle, "Shadow information spaces: Combinatorial filters for tracking targets," *IEEE Transactions on Robotics*, vol. 28, no. 2, pp. 440–456, 2012.
- [11] E. Galceran, E. Olson, and R. M. Eustice, "Augmented vehicle tracking under occlusions for decision-making in autonomous driving," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 3559–3565.
- [12] P. Ondruska and I. Posner, "Deep tracking: Seeing beyond seeing using recurrent neural networks," *ArXiv preprint arXiv:1602.00991*, 2016.
- [13] J. Dequaire, P. Ondrka, D. Rao, D. Wang, and I. Posner, "Deep tracking in the wild: End-to-end tracking using recurrent neural networks," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 492–512, 2018.
- [14] S. Brechtel, T. Gindele, and R. Dillmann, "Probabilistic decision-making under uncertainty for autonomous driving using continuous pomdps," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2014, pp. 392–399.
- [15] S. Brechtel, T. Gindele, and R. Dillmann, "Solving continuous pomdps: Value iteration with incremental learning of an efficient space representation," in *International Conference on Machine Learning*, 2013.
- [16] M. Bouton, A. Nakhaei, K. Fujimura, and M. J. Kochenderfer, "Scalable decision making with sensor occlusions for autonomous driving," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018.
- [17] S. Russell and A. L. Zimdars, "Q-decomposition for reinforcement learning agents," in *Proceedings of the 20th International Conference on International Conference on Machine Learning*, ser. ICML'03, Washington, DC, USA: AAAI Press, 2003, pp. 656–663.
- [18] L. L. Hoberock, "A survey of longitudinal acceleration comfort studies in ground transportation vehicles," *Journal of Dynamic Systems, Measurement, and Control*, vol. 99, no. 2, pp. 76–84, 1977.