Trajectory Optimization and Situational Analysis Framework for Autonomous Overtaking with Visibility Maximization

Hans Andersen[1], Javier Alonso-Mora[2], You Hong Eng[3], Daniela Rus[4] and Marcelo H. Ang Jr.[1]

Abstract— In this paper we present a trajectory generation method for autonomous overtaking of unexpected obstacles in a dynamic urban environment. In these settings, blind spots can arise from perception limitations. For example when overtaking unexpected objects on the vehicle's ego lane on a two-way street. In this case, a human driver would first make sure that the opposite lane is free and that there is enough room to successfully execute the maneuver, and then it would cut into the opposite lane in order to execute the maneuver successfully. We consider the practical problem of autonomous overtaking when the coverage of the perception system is impaired due to occlusion. Safe trajectories are generated by solving, in real-time, a non-linear constrained optimization, formulated as a receding horizon planner that maximizes the ego vehicle's visibility. The planner is complemented by a high-level behavior planner, which takes into account the occupancy of other traffic participants, the information from the vehicle's perception system, and the risk associated with the overtaking maneuver, to determine when the overtake maneuver should happen. The approach is validated in simulation and in experiments in real world traffic.

I. INTRODUCTION

Autonomous vehicles offer potential for additional safety, increased productivity, greater accessibility, better road efficiency, and positive impact to the environment. And providing mobility on demand as a service with a fleet of autonomous vehicles has added potentials such as higher throughput, better vehicle utilization, reduced number of vehicles on the road, less congestion and travel time, better accessibility, and lower cost [1].

The problem of urban driving is both interesting and difficult because it encompasses increased environmental complexity with respect to highway driving. Therefore, decision making for autonomous driving in urban areas has seen a lot of interest in recent years. Recent surveys by Paden et al. [2], Pendleton et al. [3], and Schwarting et al. [4] describe different approaches with their strengths and limitations.

Reacting to potentially hazardous unexpected situations is one of the key issues in autonomous driving in an urban environment [5]. An example scenario that we encounter very frequently during our autonomous vehicle deployment at the One-North area in Singapore is depicted in Fig.(1). In this

Fig. 1: Unexpected static obstacle in the form of an illegally parked car on a two-way street

scenario, a car is illegally parked on the vehicle's ego lane and therefore has to be overtaken. However, as this is a two-way traffic, the overtaking implies that the vehicle invades to the opposite lane, and therefore will take the traffic head-on, causing a safety hazard. In this case, a human driver may have to move slightly into the opposite lane in order to clearly see what is in front of the car ahead. Once he has gathered enough information about the road ahead, then he can safely overtake.

The problem of overtaking unexpected obstacles on a two way street has been discussed in our earlier work [6]. We formulated the problem as a constrained optimization and computed a locally optimal trajectory in a receding horizon manner. In this paper, we address further issues that may arise when attempting such safety critical maneuvers. A comprehensive situational analysis of the scenario is therefore important to obtain a safe planning framework capable of tackling such situations. Furthermore, in this extended version we validate our approach in real experiments in urban roads.

There are three main issues that we address with the situational analysis framework:

- Do we have enough time to safely complete the overtaking maneuver?
- How much information do we need before deciding whether to commit to the overtaking maneuver?
- What is the latest time when we have to make the decision on whether to commit to the overtaking maneuver or return to the ego lane?

To answer these questions, the main contributions of this paper are:

- A Model Predictive Control formulation that maximizes the amount of information that the autonomous vehicle

[1]Hans Andersen and Marcelo H. Ang Jr. are with the National University of Singapore, Singapore hans.andersen@u.nus.edu, mpeangh@nus.edu.sg

[2]Javier Alonso Mora is with the Delft University of Technology, the Netherlands J.AlonsoMora@tudelft.nl

[3]You Hong Eng is with the Singapore-MIT Alliance for Research of Technology, Singapore youhong@smart.mit.edu

[4] Daniela Rus is with the Massachusetts Institute of Technology, Cambridge, MA, USA rus@csail.mit.edu

gains along its trajectory. This allows the vehicle to make a more informed and therefore safer decision before overtaking the obstacle.

- A deterministic decision making framework for behavior planning that takes into account the risk, information sufficiency, and reachability associated with the planned trajectory. The proposed state machine enables overtaking in situations with limited perception coverage.
- Experimental results in simulation and real traffic scenarios, that demonstrate the capabilities of the algorithm in generating safe trajectories for autonomous overtaking in an urban environment.

The remaining of this paper is organized as follows. Related works are presented in Section II. Section III gives an overview of our method, followed by the state machine formulation in Section IV. The trajectory planner method is then described in Section V. The road occupancy, information sufficiency and vehicle reachability analysis are discussed in Section VI. The simulation results are discussed in Section VII, and in experimental results VIII, we demonstrate the capability of the planner in scenarios that require the ego car to break the traffic rule in order to proceed over illegally parked vehicle on a two-way street. Section IX concludes this paper.

II. RELATED WORKS

A practical application of motion planning is autonomous overtaking, which has also been widely researched in the literature. Sampling-based methods, such as Rapidly-Expanding Random Trees (RRT) and its variants [7], are popular for trajectory planning. Their strength is probabilistic completeness. However, probabilistic motion planning suffers from inherent accuracy due to discretization limits, and the computational complexity that rises exponentially as the dimensionality of the planning state space increases. Several approaches have been proposed in the recent literature to handle some of these issues. One notable variant is Minimum Violation RRT* (MVRRT*) [8]. where the authors express traffic rules as formulas using Linear Temporal Logic (LTL), and propose an incremental algorithm to generate a trajectory of a dynamical system that systematically picks which safety rules to violate and minimizes the level of risk involved. The system assumes static environment, and that the environment is known a priori. The proposed system also relies on the carefully designed set of rules and formulations. In contrast, our method operates in dynamic environments and can reason about risk directly.

The sampling based method of [9] has been implemented on Singapore-MIT Alliance for Research adn Technology's (SMART) autonomous vehicles and tested on both pedestrian, and urban environments [1]. Although it was successful in simple scenarios, it failed in the second scenario shown in Fig.(2), where the vehicle has to overtake multiple obstacles parked in parallel. Due to occlusion, the system is not able

[1]A short video that showcases the planners capabilities and shortcomings can be accessed here: [10]
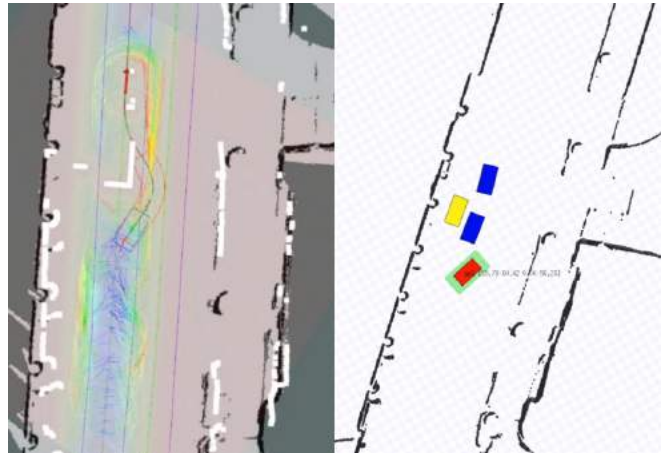

Fig. 2: Snapshot of unsuccessful overtaking path planning with probabilistic method. Left window shows robot visualization, right window shows simulation setup.

to determine an appropriate goal, and therefore reverts to a configuration at a fixed distance ahead. However, as the vehicle proceeds with the overtaking, the rest of the obstacle comes into view, and therefore the vehicle has to stop and re-plan. The planning process takes a substantial amount of processing resources and time and thus the vehicle has to be stopped. This poses a safety hazard since other vehicles may come from the other side of the lane. Furthermore, the vehicle may get stuck and not find an appropriate path after committing to the overtaking maneuver, as shown in the video.

A framework based on inverse reinforcement learning and Gaussian process to address this issue was proposed by [11]. Real world data collected from expert drivers is used to train a trajectory generator. Using the pre-trained weight, the optimal trajectory can be evaluated on-line. This approach also relies on manually defined and engineered features that have to be carefully chosen. The method also suffers from discretization error due to discontinuity in the problem formulation and training. In general, learning based motion planning methods often act as black boxes that are very difficult to systematically analyze and therefore prove safety. Acceptable driving styles under unexpected situations can differ from one place to another, and therefore a network that has been trained under one circumstance may not be applicable in the other.

Several recent works aimed at addressing the issue of limited visibility in autonomous driving. [12] proposed a method to abstract dynamic objects and static obstacles as time dependent geometric bodies, and used optimization to compute the reachable trajectory based on this information. [13] considered the problem of driving through blind intersection. The authors proposed a model that considers the sensor coverage of the vehicle, and used optimization methods to compute speed limit of the vehicle. In contrast, our method considers visibility as a part of both decision making and trajectory generation, such that the planned

trajectory is safe and maximizes the information gain.

The problem of information sufficiency is often addressed in the literature by geometric methods. For instance, Davis et al. [14] proposed a method for quantifying coverage sensing uncertainty of a robot, and uses this to plan locally optimal coverage path. Roelofsen et al. [15] proposed a reciprocal collision avoidance algorithm that guarantees collision avoidance when the robots are only capable of sensing the environment with limited field of view and Richter and [16] proposed a learning based method to plan for visibility in unknown environment.

In general, risk assessment can be broadly grouped to two categories: probabilistic and deterministic. Dou [17] used a gated branch neural network to probabilistically model lane changing behaviour on highways. Probabilistic methods take into account uncertainties that are present in the system. This approach is less conservative compared to deterministic approaches, but it can be computationally expensive, and prone to modeling error that may lead to overly aggressive and unpredictable behavior.

Deterministic risk assessment approaches on the other hand are relatively simpler and computationally efficient. For example, [18] performed formal verification of autonomous vehicle trajectory using reachability analysis. [19] used reachability analysis for safe behavior in autonomous vehicle convoy. Deterministic approaches can be prone to uncertainties, therefore the analysis has to be performed in a more conservative way.

However, the above mentioned works have not addressed all of these issues together, i.e. the problem of information and reachability in a holistic way. We believe that this is an important issue that has to be addressed in order to manage the risk without being too conservative. In this paper, we take the more conservative approach and use deterministic risk assessment. This is due to the fact that the designed behavior planner explicitly breaks the traffic rule in order to progress along the vehicle's course.

Autonomous driving in urban environments often requires smoothness and accurate modeling the car dynamics. These requirements render Receding Horizon optimization schemes, such as MPC, well suited for solving these problems. MPC has seen many applications in autonomous driving, specially in trajectory tracking applications. For example in [20] generates and tracks trajectory in highway context. It has also been applied as parallel autonomy planner, in which the autonomous system works hand in hand with a human driver [21]. Furthermore, [22] proposed an MPC based trajectory planner for autonomous driving along the Bertha- Benz Memorial Route. Static and dynamic obstacles are represented as polygons, and road boundaries are used as heuristics on which side of the obstacle to overtake from.

In this paper, we employ a MPC-based local motion planner, which captures the dynamics of the car, avoids obstacles, and consider visibility maximization, to generate overtaking trajectories that take into account the perception limitations of the ego vehicle.

III. Overview

The overall planner architecture is described in Fig.(3). The method can be briefly summarized as follows. Assuming that the lane boundaries are known a priori, the perception system will detect obstacles in both the ego lane and the opposite lane. The perception system also estimates the amount of blind spot caused by the occluding obstacle in the ego lane.

The state machine then makes a decision based on inputs from the perception system, that determines the behavior of the system. Different system behaviors, such as overtaking or remaining on the lane are specified by modifying the parameters, but not the formulation of the Receding Horizon planner. The optimizer then generates a safe trajectory by solving a non-linear constrained optimization in a MPC style. The cost and reward terms of the problem consist of:

- path following errors
- progress along the desired path
- velocity error
- size of blind spot (visible area)
- control inputs

While the constraints of the problem consist of:

- motion model of the vehicle
- collision avoidance with respect to obstacles
- maximum yaw rate
- maximum angular deviation from the path
- maintenance within the road boundaries

An off-the-shelf non-linear optimizer is then periodically called to solve the optimization problem, and the optimal input is given to the system.

IV. Behavioral Planner

A finite state automata, shown in Fig.(4) has been designed for behavioral level planning. The behavior planner has 5 states and 11 transitions.

Formally, we can write the state machine as $M = (\mathcal{S}, \Sigma, \nu, s_0)$ where
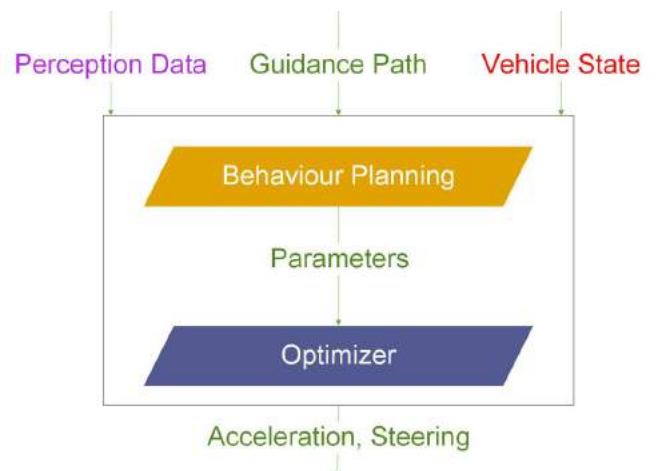
- $\mathcal{S}$ is a finite set of states.

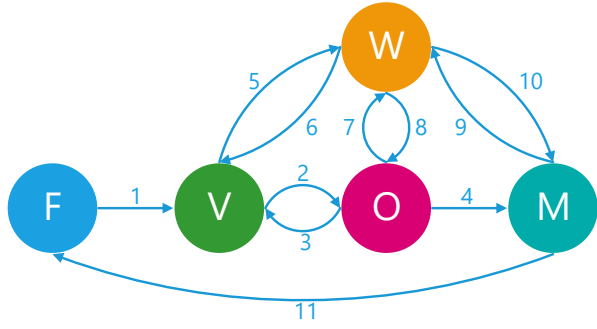Fig. 3: Overall planner architecture

Fig. 4: Behavior planning finite state machine diagram

- Σ is the state machine alphabet.
- $\nu : \mathcal{S} \times \Sigma \to \mathcal{S}$ is the transition function.
- $s_0 \in \mathcal{S}$ is the start state.

The set of states consists of $\mathcal{S} = \{\mathbf{F}, \mathbf{V}, \mathbf{O}, \mathbf{M}, \mathbf{W}\}$. Each state correspond to specific set of parameters that is used in the optimizer, and thus a different expected behavior from the planned trajectory. The state machine is initialized at $s_0 = \mathbf{F}$

The states and the possible transitions can be described as:

- **F**: *Follow ego-lane*: In this state, the vehicle follows its own course. Therefore there should be relatively high cost of path deviation.
- **V**:*Visibility Maximization*: In this state, the MPC will try to find a trajectory that maximizes the sensor visibility, and thus in the MPC formulation, the weight on the visibility cost has to be non zero, while the weight on the path deviation cost has to be reduced.
- **O**:*Overtake*: When the vehicle has committed to the overtaking maneuver, the additional guidance path has to be computed. The MPC planner will therefore plan a trajectory that follows the suggested guidance path, while complying to obstacle avoidance constraints.
- **M**: *Merge back*: In this state the vehicle merges back to its own lane, and therefore the guidance path is restored to its original guidance path.
- **W**:*Wait*: The wait state is meant to be a placeholder state where the system waits for a change in situation to reassess the appropriate action. In this case the trajectory planner will plan a comfortable deceleration trajectory to stop.

Each transition from one state to the other is triggered by specific alphabet unique to the state. The alphabet consists of $\Sigma = \{\sigma_1, \sigma_2 \dots \sigma_{11}\}$, where

- σ_1 : Obstacle to be overtaken in ego lane detected.
- σ_2 : Visibility and overtaking time is sufficient / no feasible ego lane trajectory.
- σ_3 : Complete occlusion.
- σ_4 : Overtaking maneuver is completed.
- σ_5 : Incoming traffic in opposite lane detected and overtaking time is insufficient.
- σ_6 : Incoming traffic is cleared, and sufficiency criteria

not yet fulfilled.
- σ_7 : Incoming traffic in opposite lane detected and overtaking time is insufficient.
- σ_8 : Incoming traffic is cleared, and sufficiency criteria are fulfilled.
- σ_9 : Incoming traffic in opposite lane detected and overtaking time is insufficient.
- σ_{10} : Incoming traffic is cleared, and overtaking maneuver is completed or canceled.
- σ_{11} : Merging maneuver is completed.

In the following we describe the trajectory optimization planner and the behavioral analysis.

V. TRAJECTORY GENERATION

Based on the desired behavior, a locally optimal trajectory can be generated in a receding horizon manner.

A. Vehicle Model

In this work we employ a bicycle kinematic model Fig.(5), due to the relatively low driving speed application. The proposed method could also be combined with a nonlinear combined slip vehicle model [21] for driving at higher speed. Denote by t_0 the initial planning time and by Δt_i the i-th timestep of the planner. We consider a discrete time system with time $t_k = t_o + \sum_{i=1}^{k} \Delta t_i$. The configuration of the ego vehicle at time k is denoted as $\mathbf{z}_k = [\mathbf{p}_k, \phi_k, \delta_k, v_k] \in \mathcal{Z}$, where $\mathbf{p}_k = [x_k, y_k]$ is the position, ϕ_k is the orientation, v_k is the linear velocity, and δ_k is the steering angle of the vehicle in the global frame. The control input to the system at time k is denoted as $\mathbf{u}_k = [u_k^\delta, u_k^a] \in \mathcal{U}$, where u_k^δ is the steering rate $\dot{\delta}_k$ and u_k^a is the linear acceleration a_k.
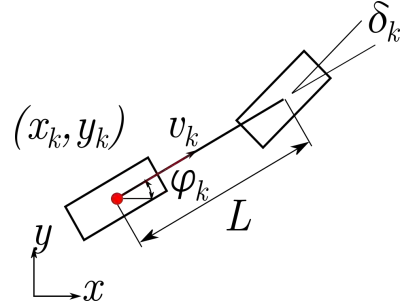

Fig. 5: Kinematic bicycle model of an Ackermann-steered vehicle.

The continuous state-space equation of the system can be written as

$$\underbrace{\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \\ \dot{\delta} \\ \dot{v} \end{bmatrix}}_{\dot{\mathbf{z}}} = \begin{bmatrix} v\,cos(\phi) \\ v\,sin(\phi) \\ (\frac{v\,tan\delta}{L}) \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} u^\delta \\ u^a \end{bmatrix}}_{\mathbf{u}} \quad (1)$$

where L is the wheelbase of the vehicle. The discrete time state space system

$$\mathbf{z}_{k+1} = f(\mathbf{z}_k, \mathbf{u}_k) \quad (2)$$

can be approximated with the integration model $\mathbf{z}_{k+1} = f(\mathbf{z}_k, \mathbf{u}_k) = \mathbf{z}_k + \int_k^{k+\Delta t} \dot{\mathbf{z}} \, dt$. In the optimizer we use fourth order Runge-Kutta integration method is used in the optimizer for sufficient accuracy.

The states of the system (steering angle $\|\delta\| \leq \delta_{max}$, longitudinal speed $v \leq v_{max}$ yaw rate $\|\dot{\phi}\| \leq \dot{\phi}_{max}$), as well as the control inputs (steering rate $\|u^\delta\| \leq \dot{\delta}_{max}$ and acceleration $a_{min} \leq u^a \leq a_{max}$) are limited to our vehicle's specifications.

B. Path Representation and Tracking

In nominal conditions the autonomous car follows centre line of the driving lane. To track the centre line while avoiding obstacles we follow [21] and formulate a Model Predictive Contouring Control (MPCC) [23] problem. MPCC optimizes the progress along the path, while considering nonlinear projection of the vehicle's position onto the desired path. The desired path contour is the centre line of the lane, and is parametrized as piecewise continuous, continuously differentiable cubic splines with multiple knots along the path.

At a given time k, the vehicle's anchor point position $\mathbf{p}_k$ tracks a continuously differentiable reference path $(x^p(\theta), y^p(\theta))$ with path parameter θ. The vehicle's progress along the path is parametrized by arc length s with $(\partial\theta/\partial s = 1)$, and can be used to approximate the change in path parameter θ by

$$\Delta\theta \approx \Delta s = v\Delta t, \tag{3}$$

and therefore for one timestep, the evolution of parameter can be approximated by

$$\theta_{k+1} = \theta_k + v_k\Delta t_k, \tag{4}$$

In the ideal case, the path parameter $\theta^p(x_k, y_k)$ should be computed in closed form inside the optimizer as the projection of (x_k, y_k) to the path. However, for computational efficiency, $\theta^p(x_k, y_k)$ is approximated by the evolution over time during the optimization process. This approximation introduces two errors, namely the longitudinal (lag) error e_k^l along the path and the lateral (contouring) error e_k^c normal to the path as shown in Fig.(6).

The lag error can be approximated by projecting the position error of the vehicle's position to θ_k along the path's tangent vector $\mathbf{t}(\theta_k)$, formally

$$
\begin{aligned}
\tilde{e}^l(\mathbf{z}_k, , \theta_k) = &-\cos\phi^p(\theta_k)(x_k - x^p(\theta_k)) \\
&- \sin\phi^p(\theta_k)(y_k - y^p(\theta_k))
\end{aligned} \tag{5}
$$

The contouring error, which measures how far the vehicle deviates from the reference path, can be approximated by projecting the position error of the vehicle's position to θ_k along the path's normal vector $\mathbf{n}(\theta_k)$, formally

$$
\begin{aligned}
\tilde{e}^c(\mathbf{z}_k, , \theta_k) = &\sin\phi^p(\theta_k)(x_k - x^p(\theta_k)) \\
&- \cos\phi^p(\theta_k)(y_k - y^p(\theta_k))
\end{aligned} \tag{6}
$$

The errors are formulated into a Receding Horizon planner (described in the forthcoming Section V) as additional cost
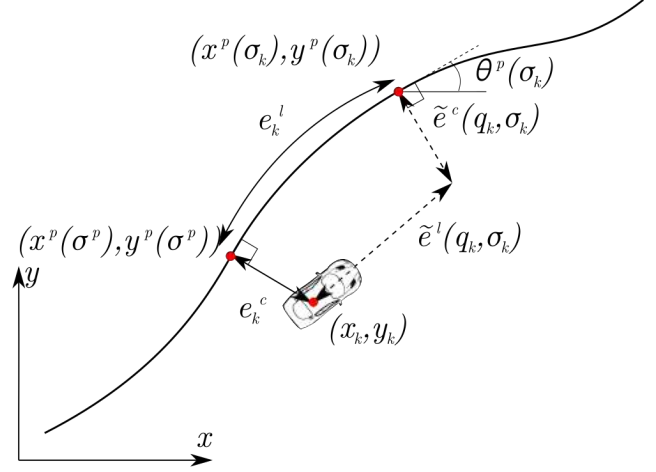

Fig. 6: Approximation of lag and contouring cost along the path.

terms, while the progress along the path is formulated as reward,

$$J_{MPCC}(\mathbf{z}_k, \theta_k) = \mathbf{e}_k^T Q \mathbf{e}_k - \gamma v_k \Delta t_k \cos(\phi_k - \phi^p(\theta_k)), \tag{7}$$

where $Q \in \mathbb{S}_+^2$ and $\gamma \in \mathbb{R}_+$ are predefined weights and the path error vector $\mathbf{e}_k$ is given by the approximated lag and contouring errors,

$$\mathbf{e}_k = \begin{bmatrix} \tilde{e}^l(\mathbf{z}_k, \theta_k) \\ \tilde{e}^c(\mathbf{z}_k, \theta_k). \end{bmatrix} \tag{8}$$

C. Road Boundaries

We follow the description of road boundaries by [21], in which the ego vehicle is represented as a union of a set of 4 circles $\mathcal{R}^j(\mathbf{z}_k, j \in \{1, \cdots, 4\})$ of radius r_{disc}. This value is chosen to enclose the vehicle's footprint, as shown in Fig.(7). The lateral distance $d(\mathbf{z}_k, \theta_k)$ of the ego vehicle's position to the reference path is given by the normal projection vector at θ^P. Since we approximate $\theta^P \approx \theta_k$, we can approximate the lateral distance by the contouring error $d(\mathbf{z}_k, \theta_k) \approx \tilde{e}^c(\mathbf{z}_k, \theta_k)$.

The drivable region at θ_k is limited by the road boundaries. The left road boundary is at distance b_l and the right road boundary is at distance b_r. To ensure that the ego vehicle drives within the limits of the road, we enforce the constraint

$$b_l(\theta_k) + w_{max} \leq d(\mathbf{z}_k, \theta_k) \leq b_r(\theta_k) - w_{max}, \tag{9}$$

where w_{max} is an upper bound of the vehicle's outline projected onto the reference path's normal. For practical purposes, it can be set as an additional padding to r_{disc}. To ensure that the vehicle can follow the desired path well in situations where road boundaries are tight, we introduce an additional constraint on the path heading difference,

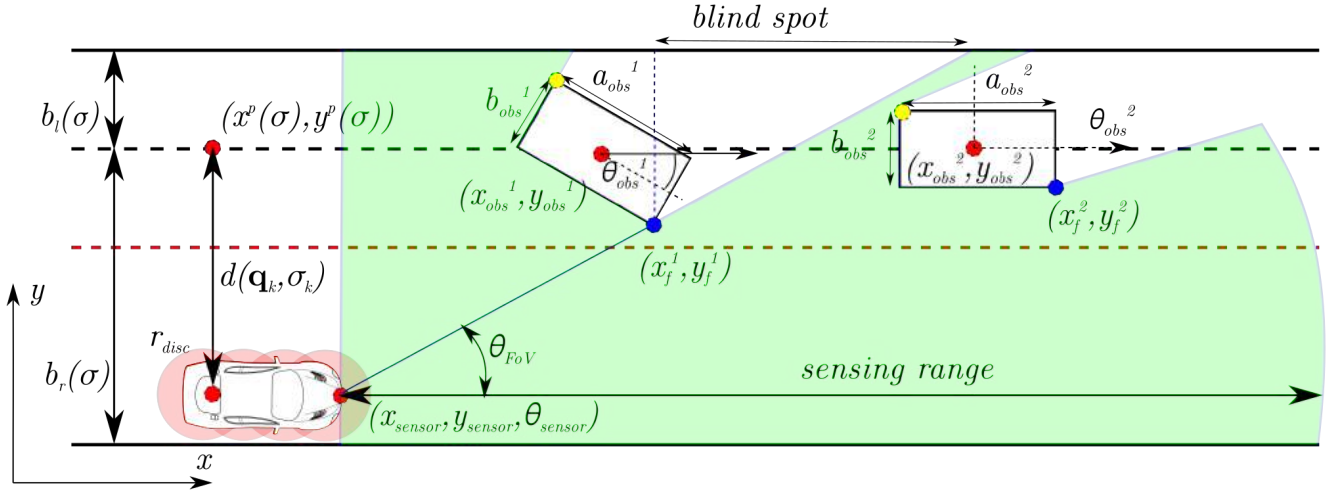$$\|\phi_k - \phi^p(\theta_k)\| \leq \Delta\phi_{max}. \tag{10}$$

Fig. 7: Figure illustrating a maneuver to increase the visibility, i.e. the area that was previously occluded.

D. Obstacle Representation

Each obstacle i, such as the wrongly parked truck, is modeled by a rectangle of length a_{obs}^i and width b_{obs}^i, such that $a_{obs}^i \geq b_{obs}^i$, whose centroid is located at (x_{obs}^i, y_{obs}^i) in the global reference frame and which has orientation ϕ_{obs}^i.

A coordinate frame is attached to each obstacle, originating at its centroid, with the x axis parallel to its length and the y axis parallel to its width. Consider the origin of the j-th circle that describes the footprint of the vehicle, its position in the i-th obstacle's coordinate frame is $x_{disc(j)}^{obs(i)}, y_{disc(j)}^{obs(i)}$. The collision constraint between the obstacle and the circle at time k can be formulated as

$$\Delta x_{k,disc(j)}^{obs(i)} = \max(\frac{-a_{obs}^i}{2}, \min(x_{k,disc(j)}^{obs(i)}, \frac{a_{obs}^i}{2})) \quad (11a)$$

$$\Delta y_{k,disc(j)}^{obs(i)} = \max(\frac{-b_{obs}^i}{2}, \min(y_{k,disc(j)}^{obs(i)}, \frac{b_{obs}^i}{2})) \quad (11b)$$

$$c_{k,disc(j)}^{obs(i)}(\mathbf{z}_k) = \frac{(\Delta x_{k,disc(j)}^{obs(i)})^2 + (\Delta y_{k,disc(j)}^{obs(i)})^2}{(r_{disc} + r_{overtake})^2} \geq 1, \quad (11c)$$

where $r_{overtake}$ is the additional safe overtaking distance.

We also employ a dynamic virtual bumper (DVB) [24] to generate a safe advisory speed v_{ref} for the vehicle. The DVB is a tube-shaped zone with its centerline as the vehicle's local path, and its width and height given by quadratic functions dependent on the vehicle's speed v_k, and the obstacles in the region.

The speed deviation is incorporated into the optimization as an additional cost term

$$J_v(\mathbf{z}_k, \theta_k) = \zeta(v_{ref} - v_k)^2, \quad (12)$$

where $\zeta \in \mathbb{R}_+$.

E. Visibility Maximization

Consider an autonomous vehicle equipped with a sensor (we use a LIDAR, but the method can be applied to other sensor types) with a limited sensing range and field of view. Now consider a scenario, when the vehicle is approaching an obstacle in a straight line section, such as the case shown in Fig.(8). We assume that the car is driving on the left side of the road and therefore has to overtake on the right side of the obstacle. Our goal is generate a motion for the vehicle such that the visibility ahead of the obstacle is maximized. For this, we first provide a definition of blind spot and then describe the cost term to be added in the Receding Horizon planner.

In the top image of Fig.(8) we show a case where an obstacle in the ego lane generates a complete occlusion along the left road boundary. For obstacle 1, we consider a frontier point (blue dot) located to the right of the centreline of the sensor located at (x_{sensor}, y_{sensor}) and with orientation (ϕ_{sensor}). We define the frontier point of a set of measurement points as the point, at position (x_f^1, y_f^1), that has the smallest field-of-view angle ϕ_{FoV}, measured from the centreline of the sensor. When complete occlusion happens, ϕ_{FoV} has negative value. The yellow point in the figure is the exact opposite of the frontier point, i.e. it has the largest field-of-view angle in the set.

As the vehicle moves towards the right boundary of the ego lane, as shown in Fig.(7) (bottom), (ϕ_{FoV}) increases. When the angle (ϕ_{FoV}) is positive, a blind spot appears, i.e. a section along the left road boundary is not covered by the sensor. This event causes a loss of information, which is necessary to determine if the overtake maneuver is safe and the autonomous vehicle can safely return to its ego lane. As seen in Fig.(7), the autonomous vehicle may need to move into the opposite lane to minimize the blind spot, and therefore maximize visibility, until it is able to make a safe decision to overtake.

Therefore, to minimize the blind spot along a trajectory we have to maximize the visibility, i.e. the field-of-view. The angle ϕ_{FoV} can be encoded into the optimization problem as a reward term:

$$J_{FoV}(\mathbf{z}_k, \theta_k) = -\lambda \phi_{FoV}, \lambda \in \mathbb{R}_+ \quad (13)$$

The weighting between visibility maximization and tracking

was tuned in simulation. Increasing the visibility maximization cost resulted in the vehicle deviating from the path earlier and more abrupt, leading to frequent wait or merge back cases when an oncoming car comes into the vehicle's sensor range. Reducing visibility maximization resulted in later and less abrupt deviation, leading to overtaking trajectories that are too late to be aborted. We tune the costs for a good tradeoff in performance.

F. MPC Formulation

We formulate a Receding Horizon planner that includes all of the aforementioned cost terms and constraints. We write the following non-linear constrained optimization,

$$\underset{u_{0:N-1}}{\text{minimize}} \quad J_t(\mathbf{z}_N, \theta_N) + \sum_{k=0}^{N-1} J(\mathbf{z}_k, \mathbf{u}_k, \theta_k) \Delta t_k \quad (14a)$$

$$\text{subject to} \quad \mathbf{z}_{k+1} = f(\mathbf{z}_k, \mathbf{u}_k) \quad (14b)$$

$$\theta_{k+1} = \theta_k + v_k \Delta t_k \quad (14c)$$

$$\mathbf{z}_{min} \leq \mathbf{z}_k \leq \mathbf{z}_{max} \quad (14d)$$

$$\mathbf{u}_{min} \leq u_k \leq \mathbf{u}_{max} \quad (14e)$$

$$\|\dot{\phi}\| \leq \dot{\phi}_{max} \quad (14f)$$

$$\|\phi_k - \phi^p(\theta_k)\| \leq \Delta\phi_{max} \quad (14g)$$

$$b_l(\theta_k) + w_{max} \leq d(\mathbf{z}_k, \theta_k) \leq b_r(\theta_k) - w_{max} \quad (14h)$$

$$c_{k,disc(j)}^{obs(i)}(\mathbf{z}_k) > 1, i = \{1, \cdots, m\},$$
$$j = \{1, \cdots, 4\} \quad (14i)$$

$$\forall k \in \{0, \cdots, N\},$$

where N is the prediction horizon and m is the number of detected obstacles.

Recalling the previous sections, the cost term is given by

$$J(\mathbf{z}_k, \mathbf{u}_k, \theta_k) = J_{MPCC}(\mathbf{z}_k, \theta_k) + J_v(\mathbf{z}_k, \theta_k)$$
$$+ J_{FoV}(\mathbf{z}_k, \theta_k) + \mathbf{u}_k^T R \mathbf{u}_k, \quad (15)$$

where $R \in \mathbb{S}_+^2$ is the control input cost, which is a design parameter.

The terminal cost is defined as

$$J_t(\mathbf{z}_N, \theta_N) = \mathbf{e}_N^T Q_t \mathbf{e}_N, \quad (16)$$

where $Q_t \in \mathbb{S}_+^2$ is the terminal cost, which is another design parameter.

In both simulation and on vehicle implementation, the MPC optimization is solved at $10Hz$, with 50 steps horizon, and $0.1s$ time step, with the code generated by FORCES Pro [25], a commercial code generator for optimization solvers, that implements interior-point methods for multistage nonlinear nonconvex optimization problems. It is noted that solving the nonlinear MPC with nonconvexity coming from collision avoidance and curvature constraints can be challenging. Due to the nonlinear and nonconvex nature of the optimization problem, the problem is prone to convergence to local minima. To speed up such convergence, the problem is initialized with the solution from the last MPC iteration.
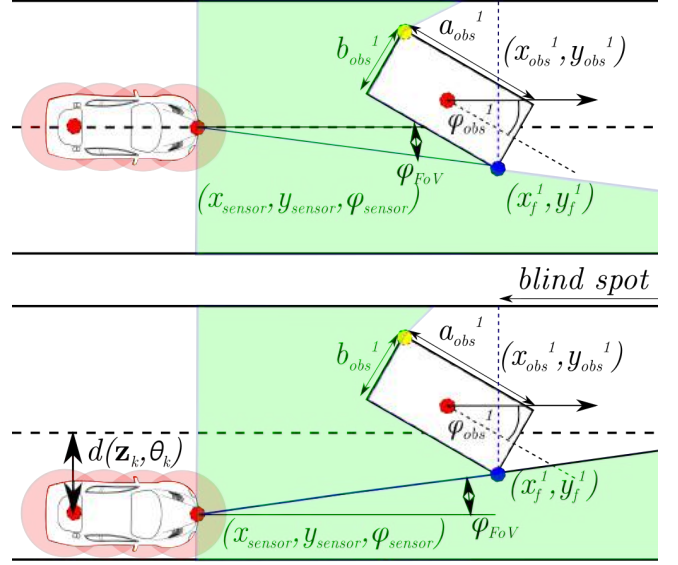

Fig. 8: Complete occlusion (top) and blind spot (bottom) of a vehicle approaching an obstacle.

It is also not possible to guarantee real time computation speed of the MPC, and therefore the solver is limited by the iterations and time budget. If the solver couldn't proceed within $10Hz$, the command from a backup path tracker that tracks the trajectory generated from previous iteration is used. The backup tracker used is the same as discussed in section VI-C.

VI. SITUATIONAL ANALYSIS FRAMEWORK

Understanding the scene within the road context is key to efficient and safe planning.

A. Occupancy of Other Traffic Participants

To determine the desired behavior of the vehicle through lane assignment we employ the Frenet frame [26]. The longitudinal dimension along the road is denoted as s, while the orthogonal direction is denoted as d. Path segments are represented using piecewise cubic splines.

Occupancy of the traffic participants can be visualized as lane occupancy diagrams, see Figs.(9-10) for different traffic scenarios and the corresponding lane occupancy diagrams. In the lane occupancy diagram, the abscissa represents time, and coordinate represents length of the road section in Frenet frame s. Blue shaded areas represent occupancy in the vehicle's ego lane. Red shaded areas represent occupancy in the opposite lane.

Fig.(9a) and Fig.(9b) show the representation of a static obstacle in the lane occupancy diagram. In Fig.(9a), an object is completely occluding the ego vehicle, if the line of sight from the vehicle's sensor to the frontier point (blue dot) does not intersect the left boundary of the road (assuming the vehicle is driving on the left side of the road). In this case, the lane occupancy cannot be determined and therefore has to be accounted as occupied when planning the trajectory. On
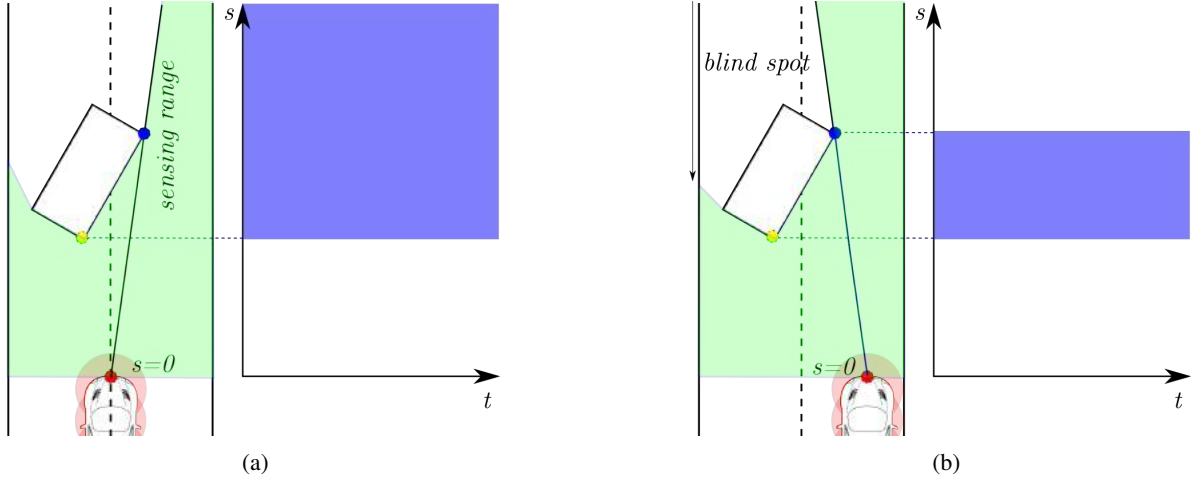
Fig. 9: Representation of (a) a complete occlusion, and (b) a blind spot by an obstacle in the lane occupancy diagram

the other hand, if the obstacle is not causing full occlusion (Fig. 9b) the occupancy of the obstacle is its length in the Frenet frame and is static from $t = 0$ until the required prediction horizon.

Due to the limitation in the sensing range, it is therefore necessary to account for the possibility of undetected incoming traffic in the opposite lane during the overtaking. This is accounted by taking the worst case scenario of a vehicle moving at the maximum traffic speed limit v_{max} at the edge of the perception. This is shown in Fig.(10), at $t = 0$ the opposite lane is assumed to be occupied at $s = $ sensing range and v_{max} is the slope of this occupancy region. When incoming traffic is detected, the assumed occupancy is no longer required and replaced by the occupancy of the traffic.

Referring to Fig.(11), the amount of overtaking time can be computed by finding the free region in the lane occupancy diagram. The occupancy set for i-th object in ego lane can be written as $\mathcal{O}^i_{ego}$, and on the opposite lane is $\mathcal{O}^i_{opposite}$. By inverting the occupancy set we get the action set $\mathcal{A}^i_{ego}$, and on the opposite lane is $\mathcal{A}^i_{opposite}$. The total action space in the corresponding lane can be written as

$$\mathcal{A}_{ego} = \bigcap_{i=1}^{n} \mathcal{A}^i_{ego} \tag{17a}$$

$$\mathcal{A}_{opposite} = \bigcap_{i=1}^{n} \mathcal{A}^i_{opposite} \tag{17b}$$

In order to determine whether it is possible to overtake the i-th obstacle in the ego lane, there should exist intersection between the action space in the opposite lane and the obstacle space in ego lane:

$$\mathcal{A}^i_{overtake} = \mathcal{O}^i_{ego} \cap \mathcal{A}_{opposite} \neq \emptyset \tag{18}$$

And the available overtaking time $t^i_{overtake}$.is the minimum time in the set $\mathcal{A}^i_{overtake}$.

B. Information Sufficiency

Getting sufficient information is important in planning the behavior of the vehicle. The trajectory planner described in
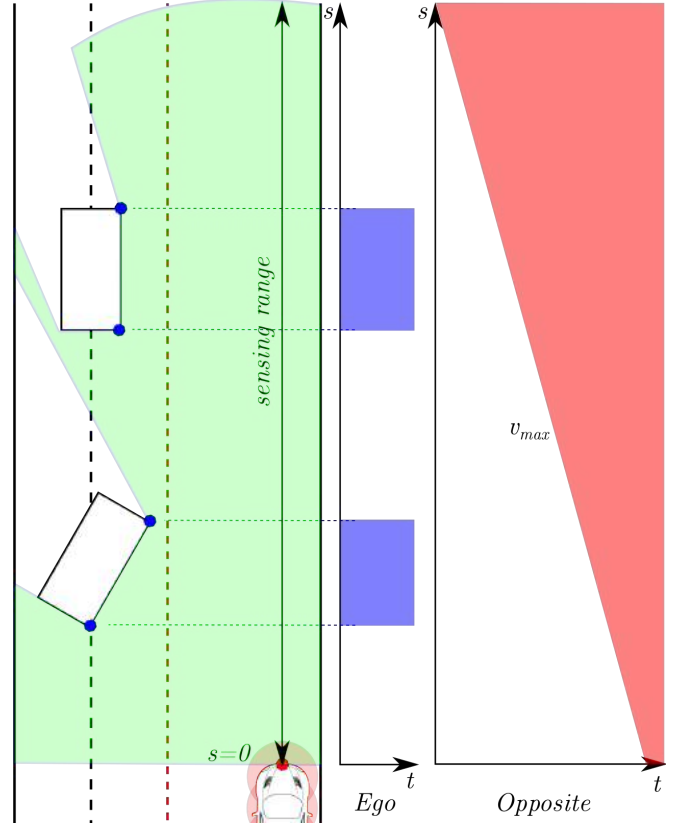

Fig. 10: Representation of incoming traffic risk in the lane occupancy diagram

section V has already taken into account the information gain in the optimization cost. It is then necessary to define a criteria for sufficient information before making further decision on whether to overtake or merge back to the lane.

An intuitive sufficiency criteria will be the minimum allowable blind spot, which is determined by the intersection between the road's left boundary and line of sight that connects the sensor with the frontier point of the obstacle.
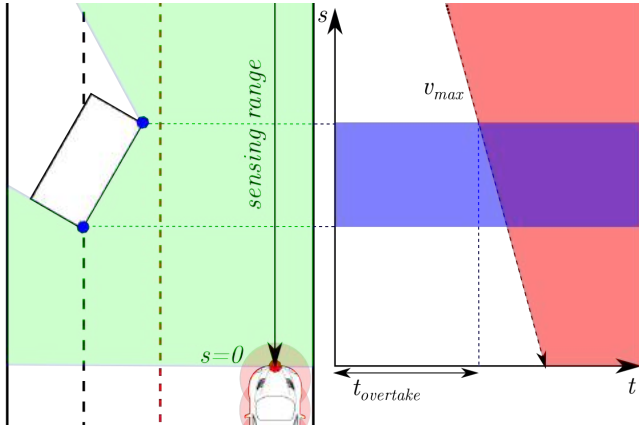
Fig. 11: Computation of available overtaking time form the lane occupancy diagram.
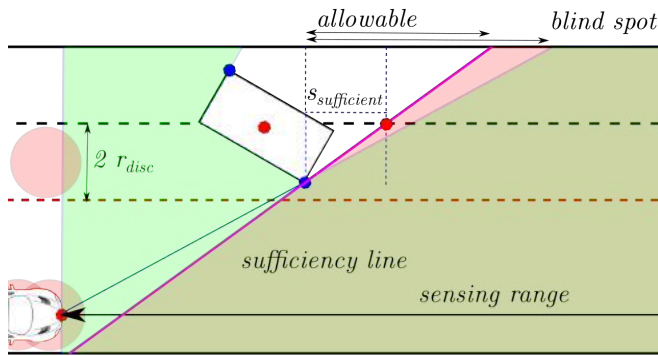

Fig. 12: Information sufficiency criteria, green area shows the sensor coverage, magenta line shows the information sufficiency boundary, $2r_{disc}$ is the accounted width of the vehicle for obstacle avoidance.

However, it is often not necessary to have sensor coverage of the objects on the road boundary. A more useful criteria would be amount of information needed such that the vehicle can safely come back to its ego lane.

This criteria is visualized in Fig.(12). In this case, the sufficiency criteria is parametrized as $s_{sufficient}$, which is the distance in Frenet frame along the line that is $2r_{disc}$ away from the right boundary of the ego lane, where $2r_{disc}$ is the accounted width of the vehicle for obstacle avoidance. The value of $s_{sufficient}$ determines how aggressive/ conservative the behavior will be. In this work, we select $s_{sufficient}$ to be the typical length of a vehicle ($4.0m$), to make sure that there is no parked vehicle directly in front of the obstacle.

The sufficiency line is defined as the line of sight from this point to the frontier point of the obstacle. And therefore as the sensor of the vehicle crosses the sufficiency line, the behavior planner can conclude that the information sufficiency criteria has been fulfilled

C. Overtaking Maneuver Risk Assessment

In order to assess the risk of an overtaking maneuver, it is necessary to know before committing the overtaking
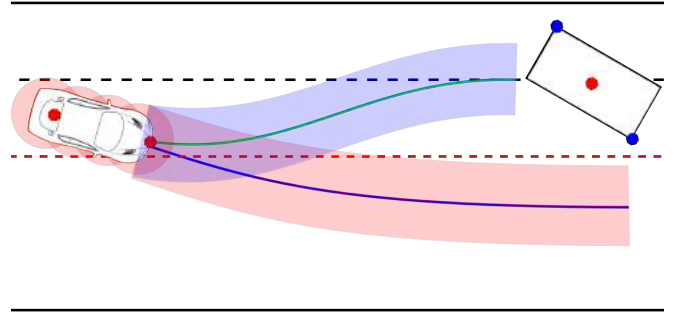

Fig. 13: Overtaking and backup trajectory generation. Blue shaded region represents the dynamic virtual bumper safety region for the backup trajectory, while the red shaded region represents the dynamic virtual bumper safety region for the overtaking trajectory

maneuver whether there is a possible trajectory that merges the vehicle back to the vehicle's ego lane. This analysis can be performed by simulating the speed profile generation and path tracking method described in [27], where maximum speed constraint is generated by considering time-to-collision in the dynamic virtual bumper. The simulation is carried with the kinematic car model with 0.1s time step. When the dynamic virtual bumper assesses that the obstacle is already too close to the vehicle or when the pure pursuit controller is not able to find a safe tracking path, the behavior planner has to decide whether to commit the overtaking path or stop and reassess the situation.

This computation is performed in parallel to the MPC based overtaking trajectory generator as shown in Fig.(13). Another advantage of computing this backup trajectory is that as the MPC optimization is highly nonlinear, there can be instances whereby the optimizer failed to generate an optimal trajectory in time, and therefore the backup command can be used instead of waiting for the next MPC plan or use the previously planned MPC solution.

Another aspect that has to be considered is the amount of time needed for the overtaking maneuver to be completed. As the length of the obstacle may not be completely known. A safe compromise would be computing the amount of time needed for the vehicle to completely pass the visible obstacles. This is performed by projecting the position extracted from the state of the MPC solution into the Frenet frame. As the time step is known, the amount of time that the vehicle needs in order to reach/ overtake the last visible s of the obstacle can be extracted.

After the behaviour planner has decided to commit the overtaking maneuver, a suggested guidance path is given to both the MPC and backup trajectory generator. This is performed by offsetting the original guidance path parallel to the lane by a distance that takes into account the obstacle width and overtaking safety margin as shown in Fig.(14). The path is reverted to the original guidance path when the

overtaking maneuver is completed and the vehicle has to merge back to its ego lane. In this way, the overall overtaking maneuver can be seen as two lane changes maneuver, and therefore the methods described here can be easily adapted to different scenarios, e.g. legal lane change.
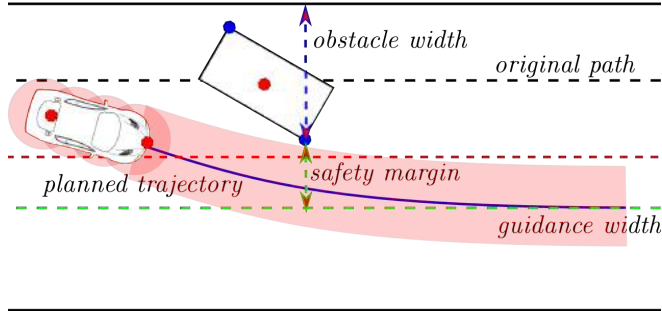

Fig. 14: Additional guidance path generation for overtaking maneuver

VII. SIMULATION RESULTS

We conduct simulations with Stage within the ROS framework. We use a previously mapped area of One North in Singapore. The simulation set up mimics the single SICK LMS-151 that has been installed on our autonomous vehicle, with 180° the field-of-view, $0,5^o$ its precision, and $50m$ the sensing range.

The simulated vehicle has a maximum cruising speed of $5m/s$. Laser scan points in the ego lane are first clustered, a rectangle is then fitted, and the blind spot frontier point extracted from the individual clusters.

A video that showcases the simulation results can be accessed at [28]. Snapshots of the different scenarios are shown in Figs. (15-17). The window of the left displays the visualization of the ego vehicle's perception and planning intentions. Purple lines indicate road boundaries, green lines indicate the center line of the path, black line indicates the MPC plan, and the red line is the backup trajectory. The MPC dynamic virtual bumper is displayed as the red polygon, and the backup trajectory dynamic virtual bumper is displayed as the yellow polygon. While white points outline the rectangular obstacle clusters in the ego lane and blue points outline the rectangular obstacle clusters in the opposite lane . The window on the top right right is the simulator. The red box is the ego vehicle and blue and yellow boxes are the obstacles. And the window on the bottom left corner is the lane occupancy diagram. Blue area shows occupancy in the ego lane, and red area shows occupancy in the opposite lane.

In the first scenario Fig.(15), the vehicle must overtake three vehicles parked in parallel, in this case the length of the obstacle is first unknown to the vehicle. When the distance to the closest obstacle is close enough, the vehicle will gather information by visibility maximization. When it gets enough information of the size of the obstacle it executes

the overtaking maneuver and safely merges back into its own lane. In this case the vehicle does not need to make use of the whole road section, as long as the information sufficiency criteria is fulfilled, the vehicle can then commit to the overtaking maneuver, and overtake within the safety distance with respect to the obstacle.
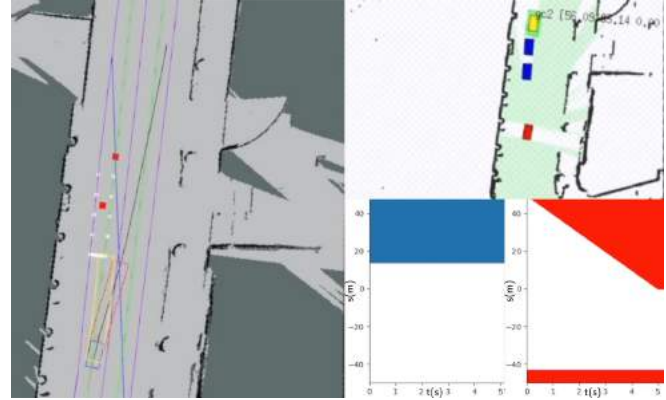

Fig. 15: Snapshot of simulation scenario 1.

In the second scenario Fig.(16), a moving obstacle is present in the opposite lane.In this case the vehicle will move to maximize visibility without invading into the opposite lane, The behavior planner then decide to let the vehicle in the opposite direction and clear the lane before proceeding with its own overtaking maneuver.
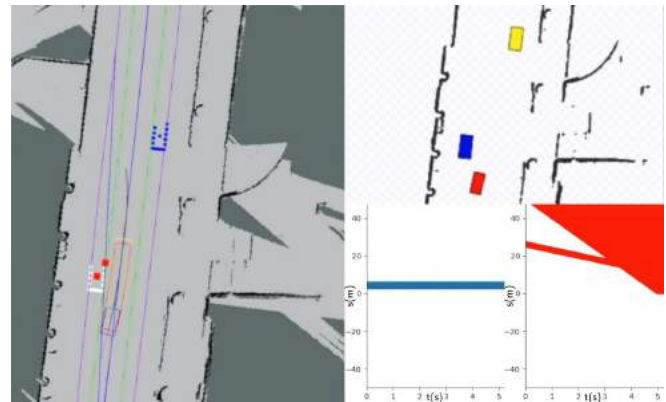

Fig. 16: Snapshot of simulation scenario 2.

In the third scenario Fig.(17), the vehicle has to overtake a slow moving obstacle int its ego lane, while reacting to a moving car in the opposite direction. In this case the vehicle will slow down and let the car in the opposite direction to pass by and clear the lane before proceeding with its own course. Another point to note is that when the obstacle in the opposite lane is detected when the vehicle already committed to the overtaking maneuver, the vehicle therefore go to the wait state, but when the opposite lane is already cleared, the sufficiency criteria are no longer fulfilled, and therefore the vehicle will transition back to visibility maximization state and restart the whole overtaking process again.
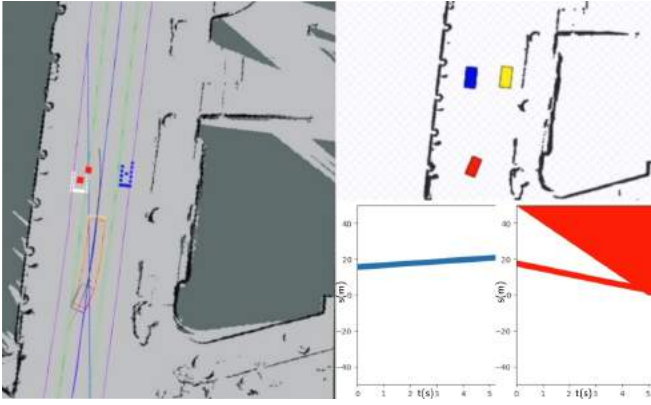
Fig. 17: Snapshot of simulation scenario 3.

Fig. 19: Snapshot of real world experiment scenario 1.

VIII. AUTONOMOUS OVERTAKING EXPERIMENTAL RESULTS

The autonomous overtaking experiment is conducted in One north, in particular the segment shown in Fig. 18 using SMART's converted iMIEV.
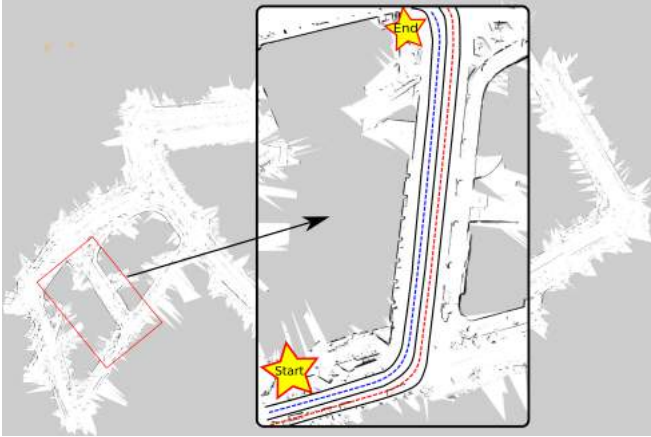

Fig. 18: Map of One North, with the particular area of interest. White pixels represent empty space, while black pixels represent vertical features extracted with the synthetic 2D LIDAR.

The start and end of the path section is marked in the inset, the centerline of the vehicle ego lane is marked with dashed blue line, while the centerline of the opposite lane is marked with dashed red line. The road boundaries are marked with solid black line. In this simulation, we assume that the road boundaries are known a priori. Road boundary information is used for two purposes: to constrain the motion of the vehicle in the MPC, and to determine whether an obstacle lies within the ego lane and/or the opposite lane.

A video showing the experimental results can be accessed at [29]. As the experiments are conducted in real world traffic, only interesting scenarios that highlight the planners capabilities are discussed here. Snapshots of the different scenarios are shown in Figs.(19, 22, 25). The window of the left displays the visualization of the ego vehicle's perception and planning intentions. The top left window shows the snapshot
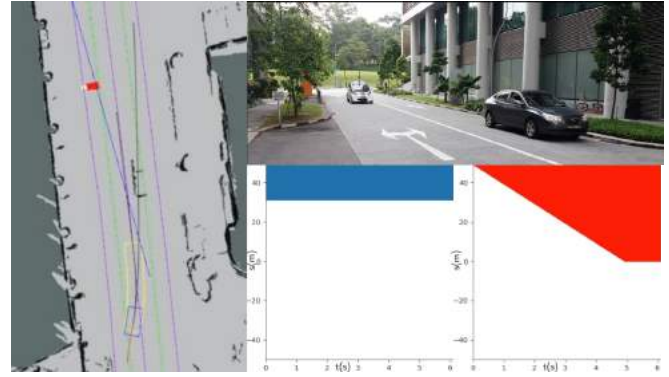
of the scenario from outside of the vehicle, and the bottom right window is the lane occupancy diagram. Figs.(20, 23, 26) show the lateral and longitudinal distances, as well as the visibility angle to the closest obstacle at different stages of the maneuver. The distances are measured are relative to the vehicle's front LIDAR frame, which is located at the front center of the vehicle. Figs.(21, 24, 27) shows the location of the vehicle, and the closest point of the obstacles in both ego and opposite lane during the overtaking maneuver. The figures also show the commanded speed constraints and the vehicle's actual speed, as well as the commanded steering wheel angle and the vehicle's steering wheel position. The speed constraint is the maximum allowable velocity that is computed by the dynamic virtual bumper safety region on the given MPC trajectory, this is the reference speed in the next iteration of the MPC.

In the first scenario Fig.(19), the vehicle simply has to overtake an illegally parked car. Similar to the simulation results in scenario 6.1, the vehicle first gathers information by visibility maximization. When it gets enough information of the size of the obstacle it executes the overtaking maneuver and safely merges back into its own lane. Throughout the maneuver the minimum lateral distance to the closest obstacle is measured at $1.7272m$, this measurement is made from the vehicle's front LIDAR. with the vehicle half width physically measured at $1.0m$, the minimum clearance to the closest obstacle is $0.7272m$.

In the second scenario Fig.(22), the vehicle starts slightly of center of the road, it tries to merge back after being overtaken by a vehicle in in the ego lane, before detecting the occluding vehicle and executing the overtaking maneuver. While executing the maneuver, the vehicle detects obstacles in the opposite lane and waits for the scenario to change and reassesses the condition before finally completing the maneuver. Throughout the maneuver the minimum lateral distance to the closest obstacle in the ego lane is measured at $1.8354m$,and therefore the minimum clearance to the closest obstacle is $0.8354m$., while the minimum lateral distance to closest obstacle in opposite lane is measured at $2.2472m$,and therefore the minimum clearance is $1.2472m$.

In the third scenario Fig.(25), the vehicle has to overtake multiple illegally parked obstacles. But, as it has already
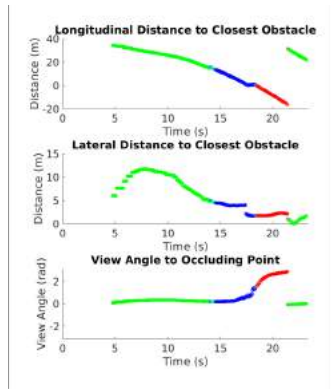
Fig. 20: Longitudinal distance, lateral distance and visibility view angle to closest obstacle plots for experiment scenario 1.
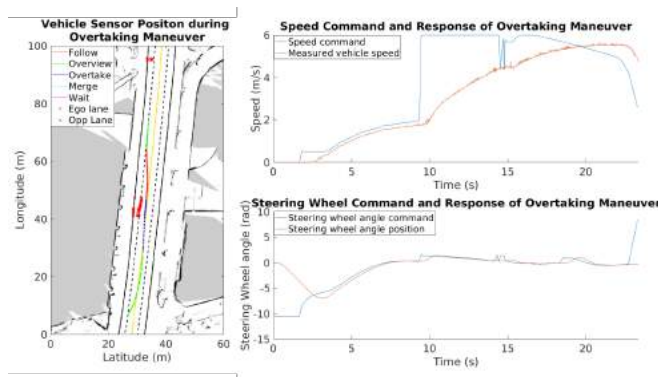

Fig. 21: Vehicle and obstacle positions, vehicle speed , and steering wheel angle plots for experiment scenario 1.
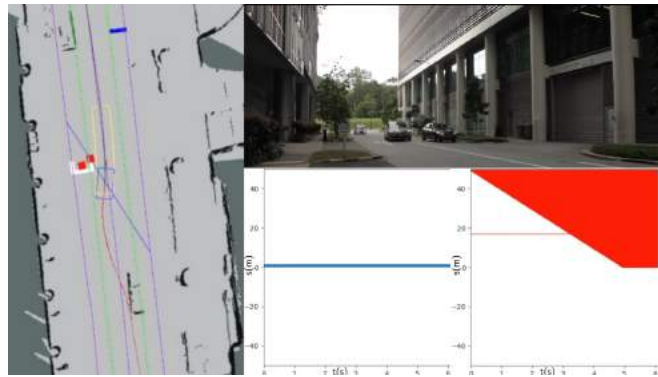

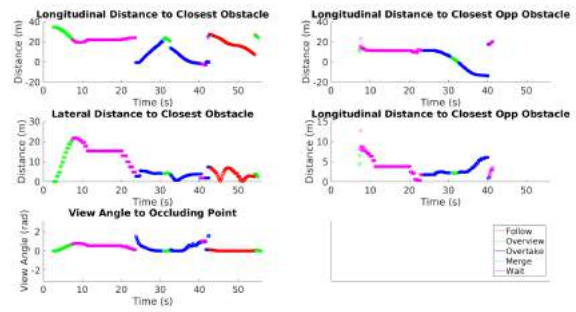Fig. 22: Snapshot of real world experiment scenario 2.

Fig. 23: Longitudinal distance, lateral distance and visibility view angle to closest obstacle plots for experiment scenario 2.
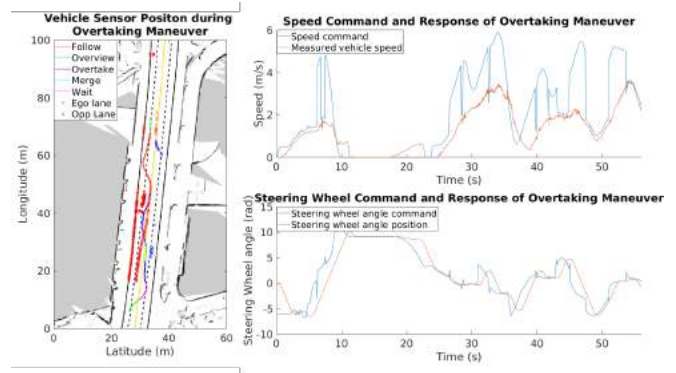

Fig. 24: Vehicle and obstacle positions, vehicle speed , and steering wheel angle plots for experiment scenario 2.

Fig. 25: Snapshot of real world experiment scenario 3.

committed to its overtaking maneuver, a jaywalker suddenly appears. The vehicle reacts by braking hard and stops for the jaywalker and safely proceed with its own overtaking maneuver. The hard brake behavior is triggered by the backup trajectory generator that senses obstacle suddenly coming into its dynamic virtual bumper and therefore its is necessary to reduce the speed rapidly. Throughout the maneuver the minimum lateral distance to the closest obstacle in the ego lane is measured at $1.1987m$,and therefore the minimum clearance to the closest obstacle is $0.1987m.$, while the minimum lateral distance to closest obstacle in opposite lane is measured at $1.5135m$,and therefore the minimum clearance is $0.5135m$.

There are potential deadlocks in the state machine, for example when the obstacle is detected very late, and reversing is needed to proceed. The current planning framework does not allow the vehicle to unstuck itself by performing a reverse maneuver or a 3-point turn, and may result in the ego vehicle waiting for indefinite amount of time. Another possible deadlock situation may occur if the incoming traffic in the opposite direction is also stuck due to the ego-vehicle's overtaking maneuver, in which human drivers would be

Fig. 26: Longitudinal distance, lateral distance and visibility view angle to closest obstacle plots for experiment scenario 3.

Fig. 27: Vehicle and obstacle positions, vehicle speed , and steering wheel angle plots for experiment scenario 3.
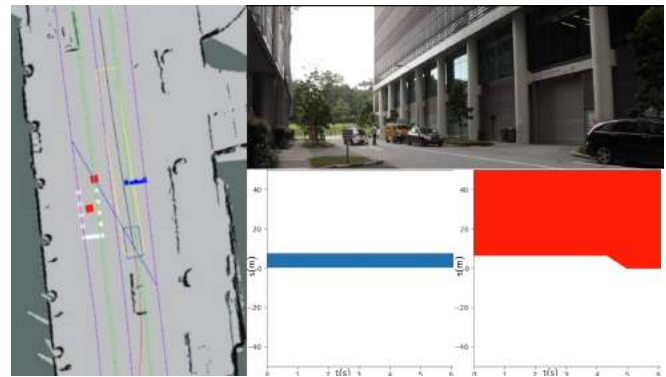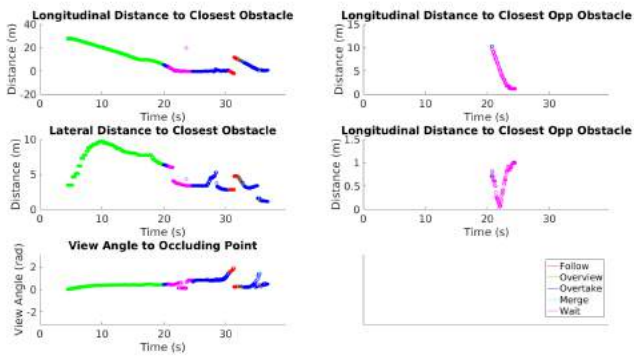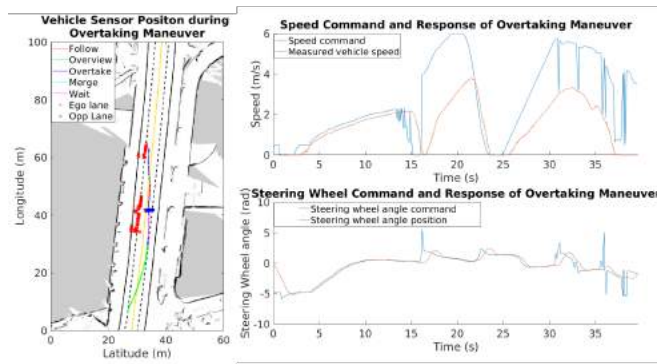
able to communicate and coordinate with each other to get unstuck.

The state machine may also enter cyclic states and is unable to proceed (i.e caught in a livelock). For example, when the system is cyclically stuck within states **V**, **W**, and **O**. A possible scenario in this case is overtaking a stop go situation, in which the vehicle's progress is blocked by the preceding vehicle. Another possible scenario is overtaking an infinitely long obstacle. e.g a jam. In such situation, the overtaking maneuver has to be aborted, and the ego vehicle has to merge back to the original preferred lane.

In the above mentioned scenarios, the role of safety driver is still important in order to make sure that the vehicle can be unstuck when such situations occur. An autonomous vehicle is a complex system that consists of many interdependent components, and removing safety driver from behind the steering wheel is still a great challenge and should be addressed in future works in order to achieve full autonomy.

IX. CONCLUSION

In this paper we have investigated the problem of overtaking unexpected obstacles on a two-way street in an urban environment. We have proposed a Receding Horizon formulation that takes into account the blind spot caused by occluding obstacles and maximizes visibility. We have also proposed a framework for analyzing the traffic situations, as well as planning the behavior of the vehicle. Our work addresses the issues of determining the amount of overtaking time available by representing the occupancy of obstacles, especially those that fully occlude the visibility of the vehicle, and may unexpectedly come from beyond the sensing range of the vehicle. Obstacle free regions are extracted from the lane occupancy diagram to determine the available overtaking time. The work in this paper also addressed the issue of information sufficiency, by determining a blind spot sufficiency criteria. The risk associated with the overtaking maneuver is addressed by determining whether a safe merge back trajectory is available and a decision is made based on a finite state machine. Simulation results and real world autonomous driving experiments have shown that the behavior planning framework is capable of handling unexpected scenarios when driving in urban environments, such as overtaking an illegally parked vehicle on a two way street.

REFERENCES

[1] Z. J. Chong, B. Qin, T. Bandyopadhyay, T. Wongpiromsarn, B. Rebsamen, P. Dai, S. Kim, M. H. Ang, D. Hsu, D. Rus, and E. Frazzoli, "Autonomy for mobility on demand," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2012, pp. 4235–4236.

[2] B. Paden, M. Cap, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 33–55, March 2016.

[3] S. Pendleton, H. Andersen, X. Du, X. Shen, M. Meghjani, Y. H. Eng, D. Rus, and M. H. Ang, "Perception, Planning, Control, and Coordination for Autonomous Vehicles," *Machines*, vol. 5, 2017.

[4] W. Schwarting, J. Alonso-Mora, and D. Rus, "Planning and Decision-Making for Autonomous Vehicles," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, no. 1, pp. annurev–control–060 117–105 157, 2018.

[5] H. Andersen, X. Shen, Y. H. Eng, D. Rus, and J. Ang Marcelo H., "Connected Cooperative Control of Autonomous Vehicles During Unexpected Road Situations," *Mechanical Engineering Magazine Select Articles*, vol. 139, no. 12, pp. S3–S7, dec 2017.

[6] H. Andersen, W. Schwarting, F. Naser, Y. H. Eng, D. Rus, and J. Alonso-mora, "Trajectory Optimization for Autonomous Overtaking with Visibility Maximization," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2017.

[7] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, Jun. 2011.

[8] L. I. R. Castro, P. Chaudhari, J. Tumova, S. Karaman, E. Frazzoli, and D. Rus, "Incremental sampling-based algorithm for minimum-violation motion planning," in *52nd IEEE Conference on Decision and Control*, Dec 2013, pp. 3217–3224.

[9] W. Liu, Z. Weng, Z. Chong, X. Shen, S. Pendleton, B. Qin, G. M. J. Fu, and M. H. Ang, "Autonomous vehicle planning system design under perception limitation in pedestrian environment," in *2015 IEEE 7th International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, 2015, pp. 159–166.

[10] H. Andersen, J. Alonso-Mora, Y. H. Eng, M. H. Ang, and D. Rus, "Autonomous Overtaking with RRT*." [Online]. Available: https://youtu.be/iwnlYLaQWLI

[11] S. H. Lee and S. W. Seo, "A learning-based framework for handling dilemmas in urban automated driving," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 1436–1442.

[12] J. Schlechtriemen, K. P. Wabersich, and K. D. Kuhnert, "Wiggling through complex traffic: Planning trajectories constrained by predictions," *IEEE Intelligent Vehicles Symposium, Proceedings*, vol. 2016-Augus, no. Iv, pp. 1293–1300, 2016.

[13] Y. Yoshihara, Y. Morales, N. Akai, E. Takeuchi, and Y. Ninomiya, "Autonomous predictive driving for blind intersections," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept 2017, pp. 3452–3459.

[14] B. Davis, I. Karamouzas, and S. J. Guy, "C-opt: Coverage-aware trajectory optimization under uncertainty," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 1020–1027, July 2016.

[15] S. Roelofsen, D. Gillet, and A. Martinoli, "Collision avoidance with limited field of view sensing: A velocity obstacle approach," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 1922–1927.

[16] C. Richter and N. Roy, "Learning to Plan for Visibility in Navigation of Unknown Environments," in *2016 International Symposium on Experimental Robotics*, D. Kulić, Y. Nakamura, O. Khatib, and G. Venture, Eds. Cham: Springer International Publishing, 2017, pp. 387–398.

[17] Y. Dou, "Gated branch neural network for mandatory lane changing suggestion at the on-ramps of highway," *IET Intelligent Transport Systems*, vol. 13, pp. 48–54(6), January 2019.

[18] M. Althoff and J. M. Dolan, "Online verification of automated road vehicles using reachability analysis," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 903–918, Aug 2014.

[19] J. Park and U. Ozguner, "Model based controller synthesis using reachability analysis that guarantees the safety of autonomous vehicles in a convoy," in *2012 IEEE International Conference on Vehicular Electronics and Safety (ICVES 2012)*, July 2012, pp. 134–139.

[20] H. Cao, S. Zhao, X. Song, S. Bao, M. Li, Z. Huang, and C. Hu, "An optimal hierarchical framework of the trajectory following by convex optimisation for highly automated driving vehicles," *Vehicle System Dynamics*, vol. 0, no. 0, pp. 1–31, 2018.

[21] W. Schwarting, J. Alonso-Mora, L. Paull, S. Karaman, and D. Rus, "Safe nonlinear trajectory generation for parallel autonomy with a dynamic vehicle model," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 9, pp. 2994–3008, Sep. 2018.

[22] J. Ziegler, P. Bender, T. Dang, and C. Stiller, "Trajectory planning for Bertha - A local, continuous method," *2014 IEEE Intelligent Vehicles Symposium (IV)*, vol. 35, no. April, pp. 450–457, 2014.

[23] D. Lam, C. Manzie, and M. Good, "Model predictive contouring control," in *49th IEEE Conference on Decision and Control (CDC)*, Dec 2010, pp. 6137–6142.

[24] S. D. Pendleton, H. Andersen, X. Shen, Y. H. Eng, C. Zhang, H. X. Kong, W. K. Leong, M. H. Ang, and D. Rus, "Multi-class autonomous vehicles for mobility-on-demand service," in *2016 IEEE/SICE International Symposium on System Integration (SII)*, Dec 2016, pp. 204–211.

[25] A. Domahidi and J. Jerez, "FORCES Professional," embotech GmbH (http://embotech.com/FORCES-Pro), Jul. 2014.

[26] E. Kreyszig, *Differential Geometry*. New York: Dover, 1991.

[27] H. Andersen, Z. J. Chong, Y. H. Eng, S. Pendleton, and M. H. Ang, "Geometric path tracking algorithm for autonomous driving in pedestrian environment," in *2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, July 2016, pp. 1669–1674.

[28] H. Andersen, J. Alonso-mora, Y. H. Eng, M. H. Ang, and D. Rus, "Autonomous Overtaking with Visibility Maximization - Experimental Results." [Online]. Available: https://youtu.be/iKXvOs6Drw0

[29] ——, "Autonomous Overtaking with Visibility Maximization - Experimental Results." [Online]. Available: https://youtu.be/iKXvOs6Drw0

Hans Andersen received B.Eng(Hons) in Mechanical Engineering, M.Sc. in Mechatronics, and Ph.D in Mechanical Engineering degrees from National University of Singapore in 2013, 2014, and 2018 respectively. His research interests include autonomous driving, motion planning, and model predictive control.

Javier Alonso-Mora is an Assistant Professor at the Delft University of Technology. Until October 2016 he was a Postdoctoral Associate at the Computer Science and Artificial Intelligence Lab CSAIL of MIT, working in the Distributed Robotics Lab. He received his Ph.D. degree in robotics from ETH Zurich, working in the Autonomous Systems Lab (2014). He holds a M.Sc. from ETH Zurich, a Diploma in Engineering and a Diploma in Mathematics from the Technical University of Catalonia. He was also a member of Disney Research Zurich. His main research interests are in navigation, decision-making, motion planning and control of autonomous mobile robots, with a special emphasis in multi-robot systems and robots that interact with other robots and humans. He received a NWO Veni award in 2017.

You Hong Eng received the B.Eng. (Hons.) and M.Eng. degrees in mechanical engineering from the Nanyang Technological University, Singapore, in 2006 and 2008, and the Ph.D. degree in industrial and systems engineering from the National University of Singapore in 2015. From 2008 to 2015, he was a Research Associate with the Acoustic Research Laboratory (ARL), Tropical Marine Science Institute, where he worked on the identification, navigation and control of unmanned underwater vehicles. He is also interested in the development of new types of actuators for underwater applications. He then joined the Singapore-MIT Alliance for Research and Technology as a Postdoctoral Associate to work on the self-driving vehicle.

Daniela Rus is the Andrew (1956) and Erna Viterbi Professor of Electrical Engineering and Computer Science and Director of the Computer Science and Artificial Intelligence Laboratory (CSAIL) at MIT. She serves as the Director of the Toyota-CSAIL Joint Research Center and is a member of the science advisory board of the Toyota Research Institute. Rus' research interests are in robotics, mobile computing, and data science. Rus is a Class of 2002 MacArthur Fellow, a fellow of ACM, AAAI and IEEE, and a member of the National Academy of Engineering and the American Academy of Arts and Sciences. She is the recipient of the 2017 Engelberger Robotics Award from the Robotics Industries Association. She earned her PhD in Computer Science from Cornell University.

Marcelo H. Ang Jr. received his Ph.D. degree in Electrical Engineering from the University of Rochester, Rochester, New York, in 1988. His work experience includes heading the Technical Training Division of Intel's Assembly and Test Facility in the Philippines, research positions at the East West Center in Hawaii and at the Massachusetts Institute of Technology, and a faculty position as an Assistant Professor of Electrical Engineering at the University of Rochester, New York. In 1989, Dr. Ang joined the Department of Mechanical Engineering of the National University of Singapore, where he is currently an Associate Professor, with a Joint Appointment at the Division of Engineering and Technology Management. He also is the Acting Director of the Advanced Robotics Centre. His research interests span the areas of robotics, mechatronics, and applications of intelligent systems methodologies.