2019 IEEE Intelligent Vehicles Symposium (IV)
Paris, France. June 9-12, 2019

A POMDP Maneuver Planner For Occlusions in Urban Scenarios

Constantin Hubmann[1], Nils Quetschlich[2], Jens Schulz[3], Julian Bernhard[4], Daniel Althoff[5] and Christoph Stiller[6]

Abstract— **Behavior planning in urban environments must consider the various existing uncertainties in an explicit way. This work proposes a behavior planner, based on a POMDP formulation, that explicitly considers possibly occluded vehicles. The future *field of view* of the autonomous car is predicted over the whole planning horizon. Both, occlusions which are generated by static as well as generated by dynamic objects are hereby considered. We use Monte Carlo sampling to generate possible future episodes that are used to derive an optimized policy. The sampled episodes consider the uncertain behavior of the known traffic participants as well as the existence probability of so-called phantom vehicles in occluded areas. By representing all possible, occluded vehicle configurations by its reachable set instead of single particles, a very efficient representation is found. Therefore, we ensure to consider all possible configurations which may drive out of the occluded area in our optimized policy. We propose a generic formulation of the POMDP problem that can be applied to various scenarios for urban driving. Its performance is demonstrated by using simulation scenarios at intersections including multiple vehicles and occlusions caused by static and dynamic objects. It is shown, that the autonomous vehicle approaches occluded areas by far less conservative than a baseline strategy which considers only the current *field of view* (*fov*). This is because various, future scenarios are already considered in the policy. In fact, we show that our planner is able to drive nearly the same trajectories as an omniscient planner would.**

I. Introduction

Various research institutions and companies are currently working on bringing level-5 autonomous vehicles on the road ([1]). The expectations of such autonomous systems are ranging from creating more efficient, more comfortable and safer transportation systems up to a full revolution of urban traffic.

A major component of such a system is the decision making layer which must choose the optimal behavior under the uncertainty of the current scenario. Urban scenarios are hereby especially challenging, as their nature is highly probabilistic. An actual scene may evolve into various different, possible future scenarios.

This is because various internal states of the environment are hidden and cannot be measured directly. For example the unknown path of the other drivers, their specific driving profile or their possible interaction with the surrounding traffic [2]. Such latent state dimensions cannot be measured but only estimated. This may become even more difficult as also the measured physical state is subject to sensor noise. The authors' previous work ([2], [3]) describes a belief state planning algorithm which is able to handle such scenarios by approximating an optimal policy online. Describing the optimal solution in such scenarios with a policy instead of a trajectory allows for less conservative behavior. This is because the policy is optimized for various future scenarios and provides a respective, optimal action for possible future observations of the scene.

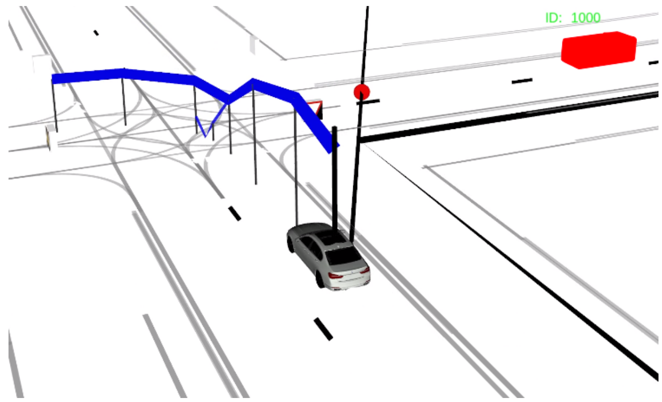


Fig. 1: A typical scenario in urban environments. The edge of an house limits the *fov* of the autonomous car. Our algorithm considers the potential *fov* over the whole planning horizon. This allows the algorithm to postpone a decision to cross to a certain point in the future where the *fov* is large enough for safe crossing. The policy offers two possible trajectories in the future, depending on what observation arrives.

The uncertainty about the surrounding traffic is again increased by the fact, that other vehicles may not be detected by the autonomous vehicle. In this work, we propose a planner that considers explicitly that other agents may be occluded and may therefore not be perceived by the sensors of the autonomous vehicle. We consider occlusions that are generated by *static* or *dynamic* objects. The potentially existing vehicles in occlusions will be referred to as phantom vehicles throughout this work.

Handling the general possibility of a possible occurrence analytically is rather difficult. Simply assuming a potential vehicle is infeasible as the worst case position of a possible other car is dependent of the trajectory of the ego vehicle which is yet to be found. Another idea would be to limit the longitudinal velocity in a way that immediate breaking before the occlusion is always possible. Nonetheless, this leads in many cases to a full standstill as the allowed velocity approaches zero while the distance to the occlusion

[1,2,3,5] are with BMW Group, D-85716 Unterschleissheim, Germany. (constantin.hubmann@bmw.de, nils.quetschlich@bmw.de, jens.schulz@bmw.de, daniel.althoff@bmw.de)
[4] is with fortiss GmbH, Research Institute of the Free State of Bavaria, bernhard@fortiss.org
[6] is with the Institute of Control, Karlsruhe Institut für Technologie (KIT), Karlsruhe, Germany. (stiller@kit.edu)

decreases.

In this work, a less conservative approach is introduced. The idea is to simulate the future *fov* over the complete planning horizon. Hereby, not only static occluding objects are considered. Dynamic, occluding objects are simulated forward in time and it is taken into account how they may affect the future *fov* accordingly. Potential vehicles in occluded areas are retrieved by mapping the future *fov* on the lanes of a topological map. Using an abstract, reachability-set based representation for phantom vehicles allows to represent all possibly occluded vehicle configurations on one lane with a low dimensional state.

We formulate the problem as a *Partially Observable Markov Decision Process* (POMDP) which allows to represent the current existence probability of a potentially occluded vehicle via belief states. The belief state is also used to represent the unknown routes of the surrounding vehicles and their probabilistic future configurations on these routes. Their uncertain longitudinal movement is considered by a probabilistic transition model.

The POMDP formulation allows to simulate how the belief state of the optimization problem changes upon particular performed actions. This may e.g. be driving with an offset to the lane center to maximize the *fov* which is referred to as information gathering in the context of POMDPs. The algorithm is integrated in our POMDP framework for decision making under uncertainty [2]. It is solved online on a continuous state-space by Monte Carlo sampling of possible episodes.

The main contributions of this work are as follows:

- A generic environment representation
- simulation of the *fov* over the planning horizon
- reachable set representation of the other vehicles
- probabilistic description of the ocurrence probabilities
- various simulations for intersection crossing

The remaining work is structured as follows. The next section, Sec. II, describes related work in the area of occlusion-aware approaches. In Sec. III, the approach is explained in detail. The sections Sec. IV and Sec. V provide details of the implementation and show the simulation results.

II. Related Work

Occlusion-aware planners frequently apply either reachability analysis or probablistic models.

The former approach assumes a participant *at the boundary* of an occluded area and adapts the behavior based on the reachable state of this participant. A simple heuristic adaptation of the safety distance based on a geometric detection of the occluded areas is proposed in [4] and evaluated with a mobile robot. In [5], the reachable set of states of potentially occluded vehicles is approximated using Kamm's circle and a velocity interval. The approach is evaluated with static and dynamic occlusions at an intersection. A combination of different map layers, tracking observed objects or defining unobserved regions, which is used for generating collision-free trajectories based on a worst-case assumptions is proposed in [6]. Based on real-sensor data, the authors demonstrate the validity of their approach. Overall, reachability-based approaches can ensure safety with respect to model assumptions. However, conservative driving is likely to emerge when using reachable sets.

In contrast, probabilistic approaches assume a distribution of participants *within* an occluded area and adapt the behavior based on the likelihood of interference with the other participant. A planning algorithm can more reliably handle occlusions from larger vehicles, e.g buses in the own lane based on probabilistic tracking mechanism, continuing over occluded areas [7]. Still, such an approach requires a prior detection of the object before entering the occluded area.

A POMDP as probabilistic model provides advantages with respect to interaction-aware behavior generation. [8] employs a POMDP approach to deal with a static vehicle that parks in front of a cross-walk. They model occlusions via the number of unobservable grids in the discretized state space. However, they control only the longitudinal acceleration and the pedestrian is not moving. In [9] a POMDP model is examined for merging onto a priority road while the field of view is occluded by static objects. A boolean flag indicates the visibility of the corresponding vehicle. To calculate visibility, they check for intersection between a straight line from the respective vehicle to the autonomous car and other objects. It is assumed that the number of vehicles in the occluded area is known a-priori. A scalable method to handle multiple occluded traffic participants based on utility fusion is presented in [10]. The authors solve a POMDP separately for each vehicle in a scenario. The resulting pairs of beliefs and actions are combined, either using a sum or minimum utility function. Comparing two offline methods, QMDP and SARSOP, the authors show, that the computation time scales linearly with the number of detected traffic participants. However, the optimal behavior must be precomputed offline. A learning-based approach using Deep Q-Learning is used to solve a POMDP which models static occlusions at intersections [11]. The authors use a discretized state space and color code occluded areas in the occupancy map using a ray tracing approach. The agent then infers the best strategy to deal with the hidden participants in the occluded area. However, Deep Reinforcement Learning impedes generalization to unseen scenarios and suffers from approximation errors.

We extend previous occlusion-aware approaches in various ways. We combine reachability analysis of potentially occluded objects with a probabilistic POMDP formulation. This allows to describe the probabilities of possible configurations of occluded objects without being dependent of sampling every configuration. We can adapt the probability of occluded ghost vehicles by using a traffic density for each lane. By simulating the *fov* over the whole planning horizon a less conservative and less reactive behavior is possible.

III. Approach

This work focuses on the online decision making for the ego vehicle, i.e. the generation of an optimized policy. This may be used for traversing an unsignalized intersection with

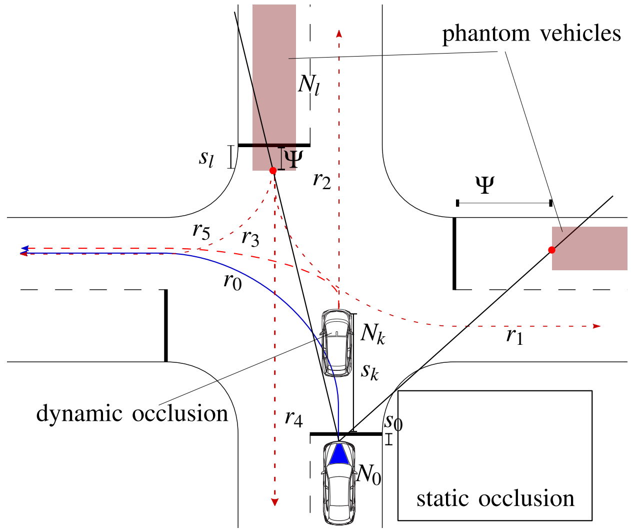


Fig. 2: This schematic shows the various variables of the state space. The vehicles are encoded by their distances to the intersection on their respective lanes. The autonomous vehicle follows its deterministic, preplanned path, r_0, while the other vehicles have various path hypotheses. The different paths also define conflict areas, which may not be occupied by the autonomous and another (phantom) vehicle at the same time. The conflict areas are not drawn for simplification issues but are explained in detail in [2].

an arbitrary layout and a variable number of other traffic participants and occluded vehicles. This section describes the general POMDP formulation first. Subsequently, the various parts of the POMDP are described in detail.

A. Partially Observable Markov Decision Process

A POMDP is defined by the tuple $\langle \mathcal{X}, \mathcal{A}, \mathcal{T}, \mathcal{O}, \mathcal{Z}, \mathcal{R}, b_0, \gamma \rangle$ with the possible states $x \in \mathcal{X}$ and possible actions $a \in \mathcal{A}$ to be executed by the agent. $\mathcal{T}(x', x, a) = P(x'|x, a)$ is the transition probability of ending in state x' when executing action a in state x. $\mathcal{R}(a, x)$ is the reward for selecting action a in state x. b_0 is the initial belief over all possible states. Additionally, the discount factor $\gamma \in [0, 1)$ is used to favor immediate rewards over rewards in the future.

The differences to a Markov Decision Process (MDP) are the possible observations $\mathbf{o} \in \mathcal{O}$ and the observation function $\mathcal{Z}$. While a MDP assumes that the current state x is given, i.e. directly observable, a POMDP assumes that the hidden state of the system can be estimated from an observation $o \in \mathcal{O}$. The observation function $\mathcal{Z}(x', a, o) = P(o|x', a)$ provides the probability to observe a certain observation o after taking action a and ending in the new state x'. Instead of generating a policy $\pi : x \mapsto a$, which maps a state to an action, the policy of a POMDP, $\pi : b \mapsto a$, maps a belief state b to an action. The belief state describes a probability distribution over all possible states. The intial belief is b_0. The solution of a POMDP is the optimal policy, π^*, that maximizes the expected, discounted cumulative reward

$$\pi^* := \arg\max_{\pi} \mathrm{E}\left[\sum_{\tau=0}^{\infty} \gamma^{t_\tau} \mathcal{R}(x^{t_\tau}, \pi(b^{t_\tau})) | b^{t_0}, \pi\right]. \quad (1)$$

B. Environment and State Space Representation

The path of the ego vehicle r_0 is assumed to be collision-free regarding static obstacles and is either generated by a path planner a priori or simply retrieved from the road geometry of a given map. Our algorithm optimizes either only the longitudinal velocity on the path (see [12] for the introduction of path velocity decomposition) or uses the path simply as a reference but optimizes the longitudinal and lateral motion in a combined manner.

The other vehicles are represented in a preprocessed state space where they must only be considered in longitudinal direction but have a belief over different possible paths, $r_i \in \mathcal{R} = \{r_1, ..., r_I\}$ for $I \in \mathbb{N}_0$, on which they are driving. This allows to consider various path hypotheses (see the previous work [2] and Fig. 2).

The state representation is defined as

$$x = (x_0, x_k, \dots, x_l, \dots)^T \quad (2)$$

with $x_0 = x_{\text{ego}}$ being the state of the autonomous vehicle N_0. The states x_k with $k \in 1, \dots, K$ describe the states of surrounding traffic N_k. The states $x_l \in [K+1, \dots, L]$ describe possible phantom objects N_l in occlusions. A phantom vehicle is placed on every route, which may intersect with the path of the autonomous vehicle, r_0, s.t. $L = |\mathcal{R} \cap r_0|$. The state of the autonomous car is defined as

$$x_0 = (s_0, d_0, v_0)^T \quad (3)$$

and the states of the other vehicles are defined as

$$x_k = (s_k, v_k, r_k)^T. \quad (4)$$

The possible phantom vehicles on other routes are described by

$$x_l = (s_l, r_l, c_l)^T. \quad (5)$$

The position of every vehicle is described by its longitudinal position s in the *Frenet* frame with its origin at the start of the intersection and on a certain route r (see Fig. 2). The autonomous vehicle may also have a deviation to the center of the preplanned path, denoted as d. The absolute velocities of all agents are described by v. The velocity of the phantom vehicles is not part of the state space, as they are always assumed with the same maximum velocity. The variable c is a boolean, indicating if a car is in the occluded area ($c = 1$) or not ($c = 0$).

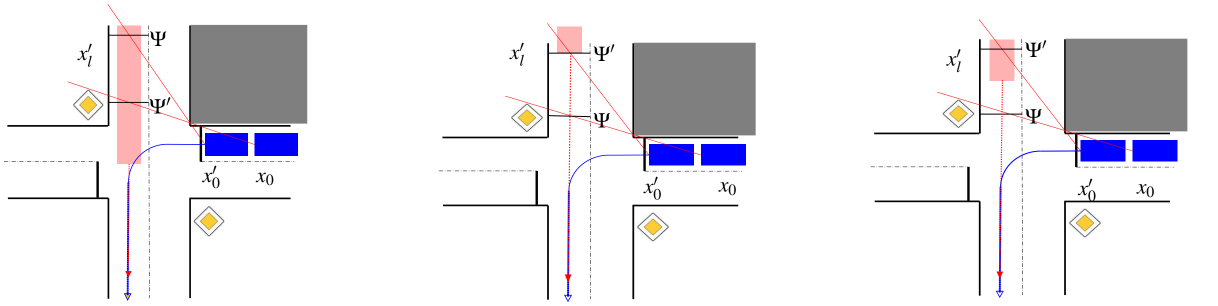
C. Action Space

We use two different discrete action sets: $\mathcal{A}_{\text{long}}$ and $\mathcal{A}_{\text{long/lat}}$. The set $\mathcal{A}_{\text{long}}$ is used in combination with a path-velocity decomposition and therefore $\mathcal{A}_{\text{long}}$ may be defined with accelerations in longitudinal direction only, s.t.

$$\mathcal{A}_{\text{long}} = \{-2\frac{\text{m}}{\text{s}^2}, -1\frac{\text{m}}{\text{s}^2}, 0\frac{\text{m}}{\text{s}^2}, 1\frac{\text{m}}{\text{s}^2}\}. \quad (6)$$

For the case of combined longitudinal and lateral actions, the non-negative longitudinal actions are combined with a set of lateral velocities $\mathcal{A}_{\text{lat}} = \{-v_{\text{lat}}, 0, v_{\text{lat}}\}$, s.t.

$$\mathcal{A}_{\text{long,lat}} = \mathcal{A}_{\text{long}} \cup (\{\mathcal{A}_{\text{long}} \geq 0\} \times \mathcal{A}_{\text{lat}}). \quad (7)$$



(a) $c = 1 \wedge s_l > \Psi_l$: A phantom car which has been already moved out of the *fov* before is set a step forward.

(b) $c = 1 \wedge y = 0$: A phantom car is sampled to not drive out of the occluded area.

(c) $c = 1 \wedge y = 1$: A phantom car is sampled to drive out of the occluded area.

Fig. 3: The three different cases which must be handled by the probabilistic transition model for the case that a phantom vehicle is there.

In this case, the planning problem changes from a longitudinal one to a combined longitudinal and lateral planning problem. We scale the possible lateral velocity with the longitudinal velocity of the autonomous car to guarantee kinematic feasibility. Therefore, v_{lat} is defined as $v_{\text{lat}} = min(0.17v_0, 0.75)$ as done in [3].

D. Observation Model

An observation $\mathbf{o} \in \mathcal{O}$ is also defined or for all three vehicle types, such that $\mathbf{o} = (o_0, o_k, \ldots, o_l, \ldots)$. Localization noise is assumed to be zero, such that the observation of the ego vehicle is a direct mapping from state to observation, such that $o_0 = (s_0, v_0)^T$.

The observation of the surrounding vehicles is defined in global coordinates, as the route is a latent variable and cannot be observed directly: $o_k = (s_k, v_k, x_k, y_k)^T$.

Please see [2] for more details on how the mapping to global coordinates and the inference of the latent variable r works.

For every potential phantom vehicle, an observation is also generated. It is simply the actual field of view Ψ on a given route. $o_l = (\Psi_l, r_l)$.

E. Representation of the unknown

Simply representing all possible vehicle configurations in the occluded area is computationally infeasible. At the same time, only considering a certain worst-case configuration is not possible. For example, simply placing a phantom vehicle at the frontier of the *fov* with maximum velocity must not be the worst case. The worst-case configuration of another car is completely dependent of the trajectory of the ego vehicle which is yet to be optimized. This coupled problem is analytically not solvable. Our idea is twofold. At first, we define the traffic density as a uniform probability distribution in the occluded area. The probability for the existence of at least one phantom car, given a discovered lane interval of a length u, is $P_p(u)$ is defined with the Bernoulli distribution:

$$P_p(u) = \begin{cases} 0 & \text{, for } u < 0 \\ \frac{u}{\omega} & \text{, for } u < \omega \\ 1 & \text{, for } u >= \omega. \end{cases} \tag{8}$$

The specific volume ω is defined as the length on which another vehicle pops up (inverse of the traffic density: number of vehicles per 100m). Given that another vehicle is inside the questioned interval, the idea is to not sample every possible configuration. Instead we represent the reachable set of all possible vehicle configurations behind the *fov*. This is realized by placing a phantom vehicle at the start of the *fov* with assumed infinite length and a velocity above the speed limit (to represent a worst-case speeding assumption), s.t. $v_{\text{phantom}} = 1.3 \cdot v_{\max}$. Given the uncertainty of the existence of another vehicle, the only safe way to cross is a sufficiently high, free *fov* at the future point in time where the crossing decision has to be made.

F. Transition Model

The motion model for the autonomous vehicle is formulated with the discrete longitudinal dynamics as

$$x'_0 = \begin{bmatrix} s'_0 \\ v'_0 \\ d'_0 \end{bmatrix} = \begin{bmatrix} 1 & -\Delta t & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_0 \\ v_0 \\ d_0 \end{bmatrix} + \begin{bmatrix} -\frac{1}{2}\Delta t^2 & 0 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} \begin{bmatrix} a_{\text{long}} \\ v_{\text{lat}} \end{bmatrix} \tag{9}$$

For the other vehicles it is defined as

$$x'_k = \begin{bmatrix} s'_k \\ v'_k \\ r'_k \end{bmatrix} = \begin{bmatrix} 1 & -\Delta t & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_k \\ v_k \\ r_k \end{bmatrix} + \begin{bmatrix} -\frac{1}{2}\Delta t^2 \\ \Delta t \\ 0 \end{bmatrix} a_{\text{k}} \tag{10}$$

The action a_k of another agent N_k is retrieved from an extended version of the Intellligent Driver Model (IDM) which e.g. also adapts to road curvatures [3].

For the phantom vehicles, various different cases arise for the transition model. The *fov* is simulated forward over the complete planning horizon s.t., every state transition (x, x') also defines a current and next *fov* for every lane: (Ψ, Ψ'). The transition model of the phantom vehicles is dependent on c and the amount of new exploration, i.e. $\Delta\Psi = \Psi - \Psi'$. A positive $\Delta\Psi$ denotes then an increased *fov* and a negative $\Delta\Psi$ denotes a decreasing *fov* on a certain lane.

Case 1: $c = 0$: In the case of a state which is representing a non-existing phantom vehicle, the transition is:

$$x'_l = (\max(\Psi_l, \Psi'_l), r_l, c_l)^T \tag{11}$$

For the assumption of an existing phantom vehicle, various possibilities exist (see Fig. 3) which are sampled with the corresponding probabilities that a phantom vehicle may occur.

Case 2a): $c = 1 \wedge s_l > \Psi_l$: The first possibility is that a phantom vehicle moved already out of *fov* (in the forward simulation). Then it is simply set one step ahead with the motion model.

$$x'_l = \begin{bmatrix} s'_l \\ r'_l \\ c'_l \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_l \\ r_l \\ c_l \end{bmatrix} - \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} 1.3 \cdot v_{\max} \cdot \Delta t \qquad (12)$$

Given that the phantom vehicle is still at the edge of the *fov*, we draw a sample y $P_p(\Delta\Psi)$ from the Bernoulli distribution.

Case 2b): $c = 1 \wedge y = 0$: If a phantom vehicle is sampled to not drive out of the occlusion, it is simply set to new field of view.

$$x'_l = (\max(\Psi_l, \Psi'_l), r_l, c_l)^T \qquad (13)$$

Case 2c): $c = 1 \wedge y = 1$: Otherwise, we simply let the phantom vehicle drive out of the occlusion. The min operator is needed to respect the cases in which the environment changes in a way s.t. the *fov* decreases:

$$x'_l = (\min(\Psi_l, \Psi'_l) - 1.3\Delta t \cdot v_{\max}, r_l, c_l)^T. \qquad (14)$$

G. Reward Model

The reward in a given state is defined as

$$\mathcal{R}(x, \mathcal{A}) = \mathcal{R}_{\mathrm{A}}(a) + \mathcal{R}_{\mathrm{V}}(v) + \mathcal{R}_C(x) \qquad (15)$$

with $\mathcal{R}_{\mathrm{A}} = -100 \cdot a^2$ and

$$\mathcal{R}_v(x) = \begin{cases} -400 \cdot |v_0 - v_{des}| & v_0 \leq v_{des} \\ -400 \cdot (v_0 - v_{des})^2 & otherwise \end{cases}. \qquad (16)$$

For a crash with a real or phantom vehicle, the reward is defined as $\mathcal{R}_{\mathrm{C}} = -20000$. In the case of combined 2-D planning, a special lateral reward, $\mathcal{R}_{\mathrm{lat}} = \mathcal{R}_{\mathrm{FOV}} + \mathcal{R}_{\mathrm{lat,acc}} + \mathcal{R}_{\mathrm{lat,off}}$, is added to the the reward functional. $\mathcal{R}_{\mathrm{FOV}}$ motivates increasing the *fov*, while $\mathcal{R}_{\mathrm{lat,acc}}$ and $\mathcal{R}_{\mathrm{lat,off}}$ punish lateral offsets or driving with lateral offset.

IV. Implementation

We solve the POMDP with the tapir framework [13] which is based on the Adaptive Belief Tree (ABT) algorithm[14]. In Fig. 4, the general idea is depicted. The belief tree is constructed by sampling episodes for 200 ms. As soon as a new belief state is discovered, it is initialized with a first value estimate. This first estimate is retrieved by a A^* search for three time steps, followed by a constant velocity roll-out. By choosing the action to expand with a Upper Confidence Bound (UCB) selection, the Q-values in the nodes converge to the optimal Q-function/policy. After 200 ms of sampling episodes, the most probable reference trajectory is retrieved from the policy and send to the trajectory planning layer. Further episodes are sampled to approximate the policy even better until a new observation arrives after 1s.

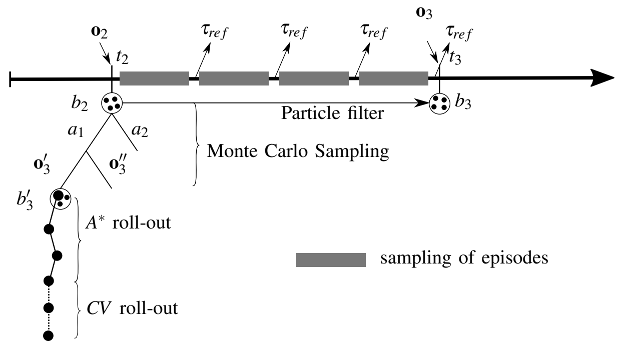


Fig. 4: Construction of the belief tree.

TABLE I: Parameter values in the final evaluation.

Parameter	Value
UCB	20000
δ_t	1 s
Computation Time	200 ms
t_{Hor}	6s
ρ	1
γ	0.8

The A^* algorithm for the roll-out is based on a heuristic. As heuristic, we use the idea of calculating the Inevitable Collision State (ICS) [15] given the prediction of the ghost vehicles.

V. Evaluation

We evaluate our approach with the values given in table I. Our approach is denoted Occlusion-aware POMDP (OCAPOP) in the following and evaluated against two other approaches. An omniscient approach serves as groundtruth, acting with the assumption of full observability of the scene (including occluded objects). We also compare our algorithm to a baseline approach, which always considers the worst case scenario, i.e. another vehicle exists behind the field of view with maximum velocity and infinite length. The baseline also uses the simplification that the field of view is not simulated over the planning horizon. While this is a very often used approached, it is only able to solve the scenarios due replanning.

A. Scenario with a Static Occlusion

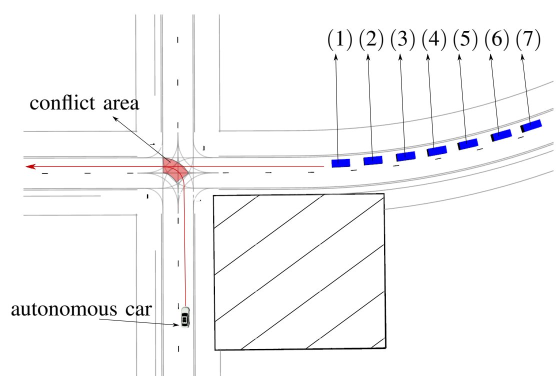


Fig. 5: An urban scenario with a static occlusion. Various possible different configurations of the other vehicle are denoted with brackets.

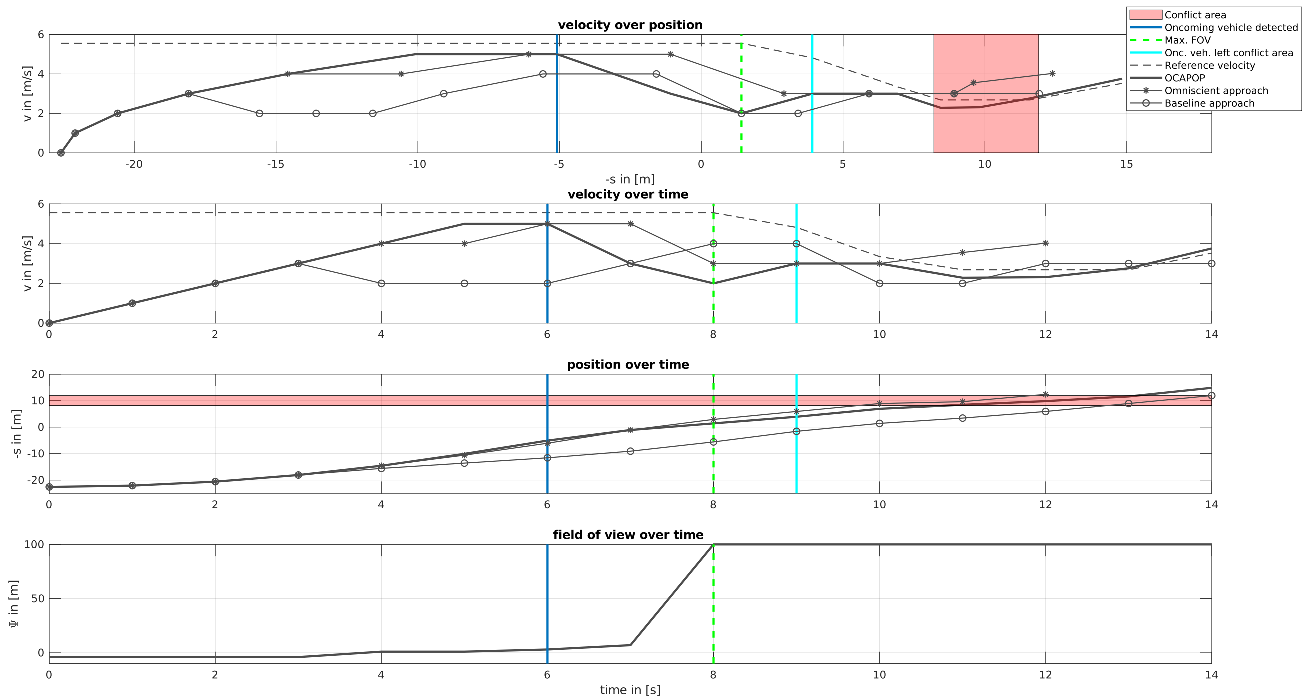


Fig. 6: Qualitative evaluation of a scenario with a static occlusion and a close oncoming vehicle.

At first a scenario with a static occlusion as shown in Fig. 5 is evaluated. We set the velocity for the maximum velocity to $v_{\max} = 20\,\frac{\text{km}}{\text{h}}$, as this emphasizes the different planning behaviors. Our evaluation is twofold. In Fig. 6 we compare OCAPOP with a baseline approach and with the omniscient planner in a qualitative way. In table II, we show the performance of our planner in a quantitative way.

1) Qualitative Evaluation: : Fig. 6 shows recorded trajectories for the scenario demonstrated in Fig. 5. Until $t = 4s$ all vehicles accelerate towards the intersection. From $t = 4s$ on, OCAPOP and the omniscient planner are able to accelerate further, while the baseline approach must already start to decelerate and drive significantly slower. This is, because the baseline planner is not able to predict the *fov* and consider stronger braking behavior in the future. OCAPOP on the other hand, acts nearly as optimistic as the omniscient approach, due its capability, to already plan several options for future observations. This behavior can also be very well seen in Fig. 1.

2) Quantitative Evaluation: : As our approach is solved by sampling of possible episodes, the generated solution is not deterministic. Therefore, every scenario is run 50 times and the results are compared in terms of the average time used to cross the intersection and in terms of the average needed acceleration. This is shown in table I. We can see that the OCAPOP performs nearly as good in terms of average crossing time and comfort as the omniscient planner, while the baseline approach is about 30% slower and has less comfort due its more reactive approach.

TABLE II: Static scenario evaluation.

Setup		Time [s]	Comfort $\sum\lvert a\rvert[\frac{m}{s^2}]$
No car	baseline	13.309	10.320
	omniscient	11.237	8.880
	OCAPOP	11.481	8.660
setup (1)	baseline	13.355	10.560
	omniscient	11.283	8.520
	OCAPOP	11.632	9.390
setup (2)	baseline	13.942	11.500
	omniscient	12.803	10.670
	OCAPOP	13.353	13.560
setup (3)	baseline	15.316	14.090
	omniscient	13.881	14.618
	OCAPOP	15.384	16.490
setup (4)	baseline	16.194	13.817
	omniscient	10.249	9.816
	OCAPOP	10.654	9.515
setup (5)	baseline	14.262	12.326
	omniscient	10.953	8.720
	OCAPOP	11.334	9.357
setup (6)	baseline	13.638	11.421
	omniscient	10.946	8.700
	OCAPOP	11.362	8.930
setup (7)	baseline	13.775	10.540
	omniscient	10.961	8.670
	OCAPOP	11.358	8.870

B. Scenario with a Dynamic Occlusion

We also present a dynamic scenario, in which the autonomous vehicle intends to turn left and the occlusion is generated by a front vehicle (see Fig. 7). table III provides the same quantitative results as for the static scenario. It can be seen, that the difference between the three planners is

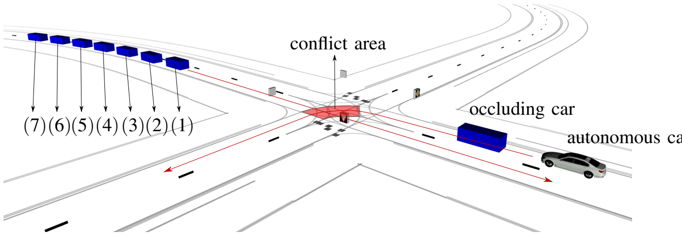


Fig. 7: A typical scenario in which the autonomous vehicle wants to turn left, but the view is occluded by another vehicle. Different possible configurations of the other vehicle are denoted in brackets.

TABLE III: Dynamic scenario evaluation.

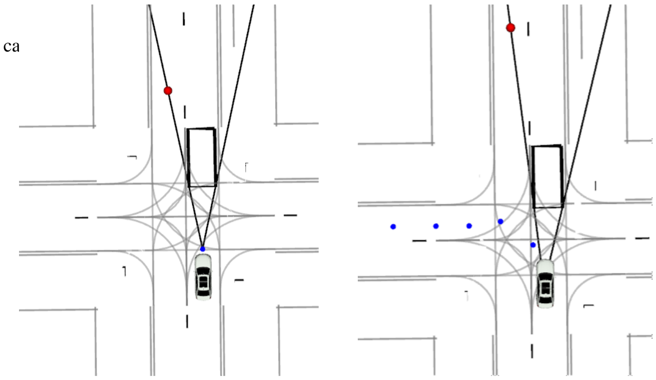
Setup		Time [s]	Comfort $\sum \lvert a \rvert [\frac{m}{s^2}]$
No car	baseline	9.505	8.600
	omniscient	9.116	9.100
	OCAPOP	9.197	9.160
setup (1)	baseline	11.382	10.100
	omniscient	11.724	11.200
	OCAPOP	11.637	11.071
setup (2)	baseline	13.638	12.960
	omniscient	13.573	13.406
	OCAPOP	13.491	12.700
setup (3)	baseline	14.274	13.074
	omniscient	10.927	10.925
	OCAPOP	12.247	11.789
setup (4)	baseline	9.194	9.024
	omniscient	9.087	9.030
	OCAPOP	9.107	8.900
setup (5)	baseline	9.430	8.745
	omniscient	9.109	9.040
	OCAPOP	9.201	8.930
setup (6)	baseline	9.584	8.630
	omniscient	9.105	9.090
	OCAPOP	9.285	8.650
setup (7)	baseline	9.638	8.660
	omniscient	9.109	9.060
	OCAPOP	9.382	8.730

not so clear anymore. This is because the vehicle in front of the autonomous car moves itself and therefore reduces the occlusion. Therefore, only very specific world configurations are solved better by our planner. This is for example the case for scenario 3, in which OCAPOP clearly outperforms the baseline.

C. Longitudinal and Lateral Optimization

We set up another scenario in which the autonomous car must actively explore the environment to enable safe crossing. Therefore, the scenario in Fig. 8 is introduced. When lateral planning is not activated, the autonomous vehicle is not able to turn left, as the *fov* is not big enough for safe traversing. For this scenario, the action vector is chosen to be $\mathcal{A}_{\text{long/lat}}$ and an additional reward $\mathcal{R}_{\text{lat}}$ is introduced to motivate lateral exploration (both are defined in Sec. III). Two snapshots of the longitudinal and the 2-D OCAPOP are shown in Fig. 8. Fig. 8a shows that also OCAPOP is not able to plan over the intersection as the *fov* is not large enough.

We compare the driven trajectory of OCAPOP again with the trajectory of an omniscient planner in Fig. 9. It can be seen that both planners drive nearly the same trajectory with the only difference of a lateral offset introduced by OCAPOP to increase the *fov*.



(a) Freezing robot, because no safe trajectory is found to overcome the occlusion.

(b) Using lateral actions, the situation can be resolved due to the increased field of view.

Fig. 8: Without lateral planning, some situations may lead to a freezing autonomous car. Using lateral planning, this can be avoided and the autonomous car safely manages the situation.

VI. Conclusion

In this work a novel approach for handling occlusions in urban scenarios is proposed. It is able to handle occlusions which are created by either static or dynamic objects in a non-conservative way. This is achieved by formulating the optimal solution as a policy for a POMDP formulation instead of generating only a simple trajectory. The idea is to propagate the *fov* over the whole planning horizon to predict how the *fov* changes over time. All possible configurations of phantom vehicles are hereby represented by a reachable set. This guarantees to not miss a possible worst-case situation. We show in various simulations, how our planner is able to drive less conservative than the baseline approach. In many cases, it even operates in the same way as an omniscient planner would. This is possible due to the policy which already respects various future possibilities given the occluded are.

The authors aim to pursue the approach. Possible future research directions would be the introduction of continuous actions and assumed, future interaction with a possible phantom vehicle.

References

[1] D. Gonzalez, J. Perez, V. Milanes, and F. Nashashibi, "A Review of Motion Planning Techniques for Automated Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, Apr. 2016.

[2] C. Hubmann, J. Schulz, M. Becker, D. Althoff, and C. Stiller, "Automated Driving in Uncertain Environments: Planning with Interaction and Uncertain Maneuver Prediction," *IEEE Transactions on Intelligent Vehicles*, 2018.

[3] C. Hubmann, J. Schulz, G. Xu, D. Althoff, and C. Stiller, "A Belief State Planner for Interactive Merge Maneuvers in Congested Trafc,"

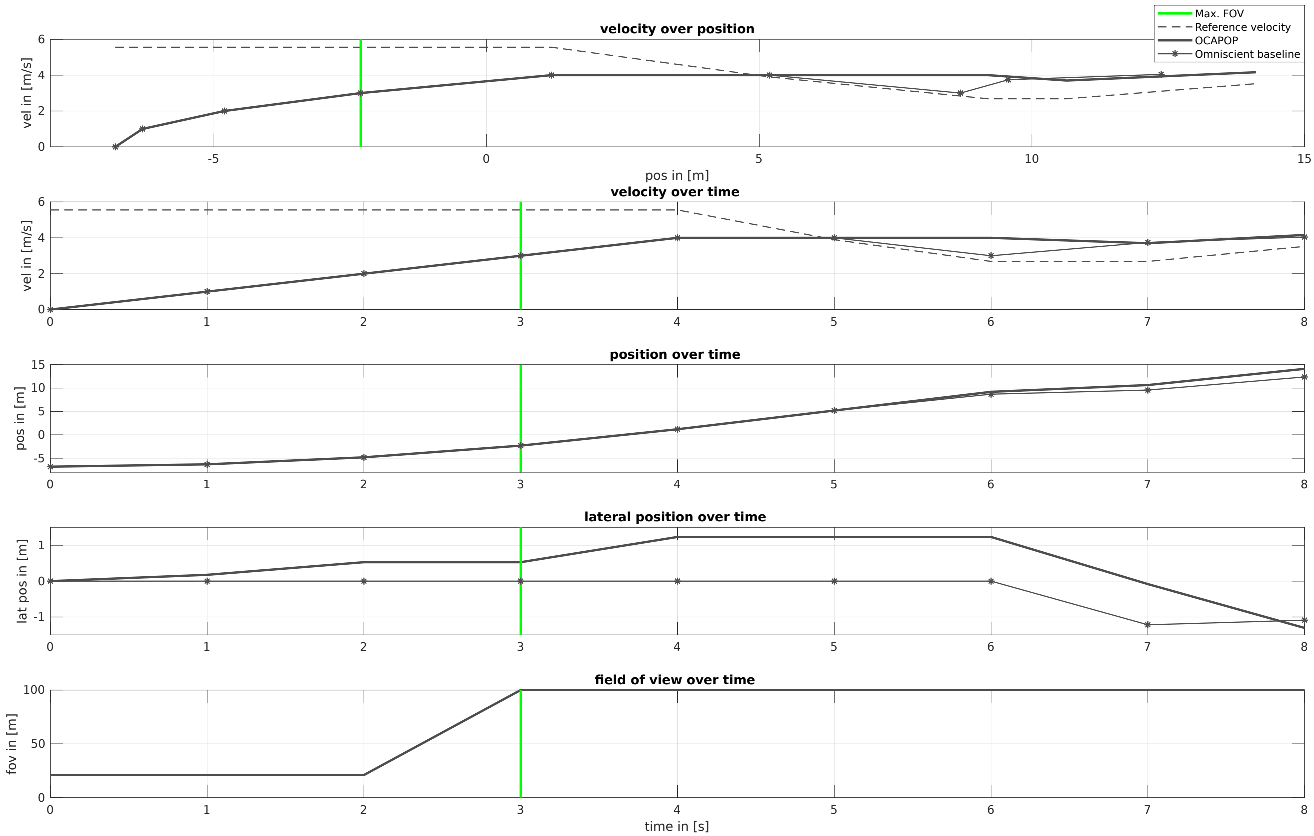


Fig. 9: Extensive evaluation for combined lateral and longitudinal planning.

[4] M. Sadou, V. Polotski, and P. Cohen, "Occlusions in obstacle detection for safe navigation," in *IEEE Intelligent Vehicles Symposium*, 2004.

[5] P. F. Orzechowski, A. Meyer, and M. Lauer, "Tackling Occlusions & Limited Sensor Range with Set-based Safety Verification," in *International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2018.

[6] S. Hoermann, F. Kunz, D. Nuss, S. Renter, and K. Dietmayer, "Entering crossroads with blind corners. A safe strategy for autonomous vehicles," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, 2017.

[7] E. Galceran, E. Olson, and R. M. Eustice, "Augmented vehicle tracking under occlusions for decision-making in autonomous driving," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference On*, 2015.

[8] S. Thornton, "Autonomous Vehicle Speed Control for Safe Navigation of Occluded Pedestrian Crosswalk," 2018.

[9] S. Brechtel, T. Gindele, and R. Dillmann, "Probabilistic decision-making under uncertainty for autonomous driving using continuous POMDPs," in *17th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2014.

[10] M. Bouton, A. Nakhaei, K. Fujimura, and M. Kochenderfer, "Scalable Decision Making with Sensor Occlusions for Autonomous Driving," *IEEE International Conference on Robotics and Automation (ICRA)*, 2018.

[11] D. Isele, A. Cosgun, K. Subramanian, and K. Fujimura, "Navigating Intersections with Autonomous Vehicles using Deep Reinforcement Learning," 2017.

[12] K. Kant and S. W. Zucker, "Toward efficient trajectory planning: The path-velocity decomposition," *The International Journal of Robotics Research*, 1986.

[13] D. Klimenko, J. Song, and H. Kurniawati, "Tapir: A software toolkit for approximating and adapting pomdp solutions online," in *Proc. Australasian Conference on Robotics and Automation*, 2014.

[14] H. Kurniawati and V. Yadav, "An online POMDP solver for uncertainty planning in dynamic environment," in *International symposium of robotics research*, 2013.

[15] T. Fraichard, "A short paper about motion safety," in *Robotics and Automation, IEEE International Conference on*, IEEE, 2007.