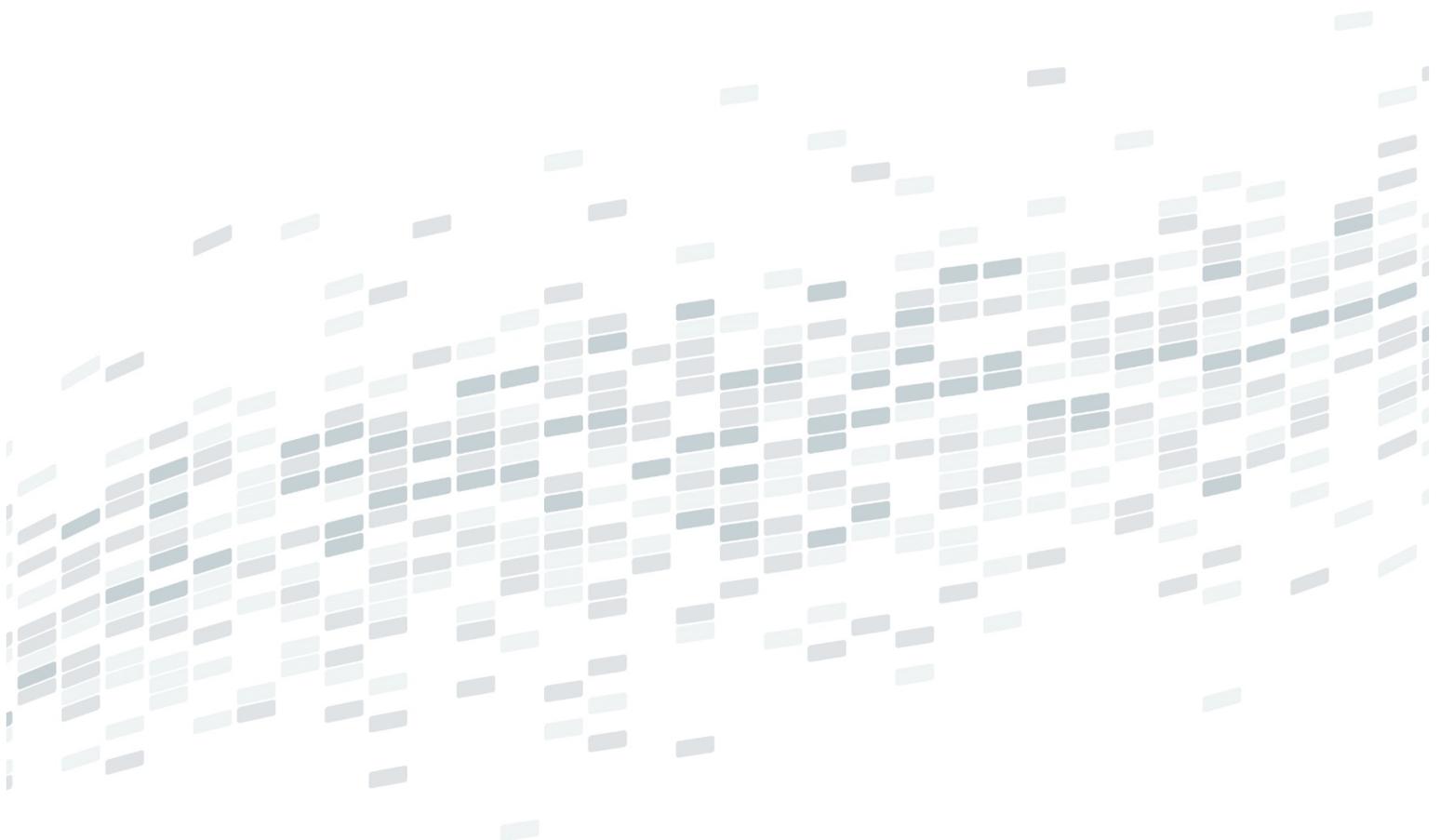


Visteon®

Towards Courteous Behavior and Trajectory Planning for Automated Driving



Towards Courteous Behavior and Trajectory Planning for Automated Driving

Oliver Speidel¹, Maximilian Graf¹, Thanh Phan-Huu² and Klaus Dietmayer¹

Abstract—Efficient behavior and trajectory planning is one of the major challenges for automated driving. Especially intersection scenarios are very demanding due to their complexity arising from the variety of maneuver possibilities and other traffic participants. A key challenge is to generate behaviors which optimize the comfort and progress of the ego vehicle but at the same time are not too aggressive towards other traffic participants. In order to maintain real time capability for courteous behavior and trajectory planning, an efficient formulation of the optimal control problem and corresponding solving algorithms are required. Consequently, a novel planning framework is presented which considers comfort and progress as well as the courtesy of actions in a graph-based behavior planning module. Utilizing the low level trajectory generation, the behavior result can be further optimized for driving comfort while satisfying constraints over the whole planning horizon. According experiments show the practicability and real time capability of the framework.

I. INTRODUCTION

In the context of automated driving, behavior and trajectory planning are basic requirements as well as major challenges. Thus, huge effort has been made in this area in recent years [1], [2], [3]. In this regard, urban scenarios still represent a vast challenge, as particularly efficient and safe behavior has to be generated. Especially when driving through intersections, the problem complexity increases dramatically, as there might be multiple merging and crossing lanes including other traffic participants. Thus, various traffic rules and the behavior of other traffic participants have to be taken into account. Traffic light guided intersections can already be handled reliably, as no complex prediction of other traffic participants is necessary [4]. In contrast, driving maneuvers without right of way at intersections or on-ramp scenarios are still demanding. For example, aggressive merging behavior may lead to fast progress of the ego vehicle but at the same time induces high costs for other vehicles as they might be forced to decelerate in order to keep a safety gap [5]. Whereas, passive behavior may lead to situations in which the automated vehicle is not able to take a turn because of congested traffic. A key element is to find a motion plan which is appropriate for other vehicles but at the same time optimizes the progress of the ego vehicle. Recent research even shows that courteous behavior leads to

better imitation of human behavior [6]. Therefore, a novel framework is presented which is able to generate trajectories optimized for comfort and progress while considering not only constraints, as for example traffic lights, but also costs for other vehicles induced by the ego trajectory. In general, a graph-based behavior planning module yields a rough trajectory which is optimized afterwards by the low level trajectory generation module.

The main contribution of this paper is two-fold. On the one hand, a novel behavior planning strategy is presented which allows to plan maneuvers under the consideration of costs for other traffic participants on a large horizon ($t_\tau \approx 10s$). On the other hand, a sampling based optimization strategy using septic polynomials and an associated replanning method is shown, which yields comfortable trajectories even in changing environments on the whole planning horizon. Thereby, the whole framework is real time capable.

II. RELATED WORK

Behavior and trajectory planning for automated driving has been widely studied. In unstructured environment a common method is to first search a drivable path and afterwards an according velocity profile is generated [1]. An approved concept for structured environment is to sample quintic polynomials in a Frénet frame which allows longitudinal and lateral movement along a given path [2], [4]. To do so, target states are determined by a rule-based heuristic depending on the current behavior state. Another approach is to use local, continuous methods [3]. However, these concepts are not well suited for planning complex maneuvers on long horizons while considering traffic rules and other traffic participants, as it is computationally infeasible [7], [8]. To overcome this problem, a novel approach is to combine behavior and trajectory planning with a graph search problem and an underlying low level trajectory optimization [7], [9]. Thereby, a rough behavior trajectory is extracted which is then used for further optimization. However, heavy sampling of quintic polynomials is necessary in order to generate smooth trajectories. Furthermore, the optimality and safety is not guaranteed on the whole horizon as in low level optimization only the next three seconds are considered. In addition, the interactions of other traffic participants are not taken into account [7], [9]. In order to plan under consideration of interactions and uncertainties, a common approach is to find the optimal policy for a Partially Observable Markov Decision Process (POMDP) [10], [11], [12]. However, these approaches are either limited in the scenarios which can be handled, lack of real time capabilities or

*This work was supported by Visteon Corporation

¹O. Speidel, M. Graf and K. Dietmayer are with the Institute of Measurement, Control and Microtechnology, Ulm University, 89081 Ulm, Germany, e-mail: firstname.lastname@uni-ulm.de.

²T. Phan-Huu is with Visteon Corporation, Karlsruhe, Germany, e-mail: tphanhuu@visteon.com

only yield discretized actions without consideration of low level optimization [10], [11], [12]. Furthermore, there are concepts accounting for costs of other traffic participants. For example, trajectories can be sampled using quintic polynomials and subsequently rated, where the according reaction of other traffic participants are considered with the Intelligent Driver Model (IDM) [5]. However, the sampling heuristics are restricted to a rule-based strategy and therefore, generating trajectories on long horizons is not practicable.

A method for producing courteous behavior is to use a game-theoretic interaction model [6]. The problem is modeled in a way that all agents try to optimize their own behavior. By predicting and considering the reaction to the ego trajectory, courteous behavior can be generated. It is shown that this courtesy leads to better imitation of human behavior [6].

For this reason, in this work the game-theoretic problem formulation is borrowed and included into a novel planning framework in order to enable courteous driving for practical use in real time.

III. PROBLEM STATEMENT

In general, the motion planning problem for automated vehicles consists of traveling towards a defined goal state in an efficient and convenient manner while obeying traffic rules. For the presented approach, it is assumed to have a given route extracted from a high-precision digital map. A common approach is to formulate the optimal control problem in Frénet coordinates and therefore it is possible to reduced the problem to find a longitudinal motion plan along the predefined center line [2]. Thereby, existing definitions can be used but have to be adapted [7]. To satisfy vehicle kinematics, acceleration bounds $a_e \in [a_{\min}, a_{\max}]$ and velocity bounds $v_e \in [0, v_{\max}]$ are introduced. Where the maximum velocity $v_{\max} = f(\kappa(s), \mathcal{R})$ is restricted by traffic rules \mathcal{R} and the curvature κ of the longitudinal position s on the given path. Additionally, constraints due to traffic lights or other vehicles can be represented as a combination of position and time intervals $\mathbf{c} = [t_{\text{start}}, t_{\text{end}}, s_{\text{start}}, s_{\text{end}}]$. Hereby, s_{start} defines the start of the spatial constraint on the center line. The end of the restricted zone is described by s_{end} . Accordingly, t_{start} and t_{end} describe the time interval in which the defined zone on the center line is forbidden.

The optimal control problem which optimizes the progress and comfort along the center line is given by,

$$\mathbf{u}^{e*} = \arg \min_{\mathbf{u}^e} J^e(\mathbf{X}, \mathbf{u}^e, \kappa(s^e), \mathbf{C}^e), \quad (1)$$

where J is a cost function, \mathbf{X} represents the current state, $\mathbf{u} = [u_k, \dots, u_\tau]$ is a sequence of actions, \mathbf{C} a set of spatio-temporal constraints and e indicates the ego vehicle. In order to enforce courteous behavior, the costs for other vehicles induced by the ego vehicle have to be taken into account.

To model a system with multiple agents, the game-theoretic optimal control problem presented in [6] is chosen and adapted. Thereby, the interaction model assumes agents optimizing their own behavior. In order to make the problem

computational tractable, a simplification is performed by approximating the behavior of other vehicles by a reactive driver model. Therefore, the action function of other vehicles is approximated by

$$\begin{aligned} \mathbf{u}^{o*} &= \arg \min_{\mathbf{u}^o} J^o(\mathbf{X}, \mathbf{u}^e, \mathbf{u}^o, \kappa(s^o), \mathbf{C}^o) \\ &\approx g^o(\mathbf{X}, \mathbf{u}^e, \kappa(s^o), \mathbf{C}^o), \end{aligned} \quad (2)$$

where o indicates other vehicles. As a result, the generic action function for the ego vehicle can be extended to

$$\mathbf{u}^{e*} = \arg \min_{\mathbf{u}^e} J^e(\mathbf{X}, \mathbf{u}^e, \kappa(s^e), \mathbf{C}^e, g^o(\mathbf{X}, \mathbf{u}^e, \kappa(s^o), \mathbf{C}^o)), \quad (3)$$

where $o \in 1, \dots, m$ and m is the number of other relevant vehicles. This formulation enables courteous behavior planning. In order to be able to plan complex maneuvers this problem has to be solved on a large horizon ($t_\tau \approx 10s$). For this reason, the following framework is presented.

IV. COURTEOUS MOTION PLANNING

In order to solve the problem formulation shown in Equation 3, a two staged optimization is used consisting of a graph based high level behavior planning method followed by a low level optimization module using septic polynomials. The general structure is based on [7]. However, the problem formulation for behavior planning and especially the low level optimization of the trajectories differs drastically from previous work.

A. Behavior Planning

The behavior planning module aims to find a rough trajectory \mathcal{T}_B which consists of discrete states \mathbf{x}_k^e , where $k = 1, \dots, \tau$, with the temporal spacing Δt . In contrast to [7], the transition model between states or respectively vertices is formulated with constant jerk (CJ) according to

$$\underbrace{\begin{bmatrix} s_{k+1}^e \\ v_{k+1}^e \\ a_{k+1}^e \end{bmatrix}}_{\mathbf{x}_{k+1}^e} = \begin{bmatrix} 1 & \Delta t & \frac{1}{2} \Delta t^2 \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_k^e \\ v_k^e \\ a_k^e \end{bmatrix} + \begin{bmatrix} \frac{1}{6} \Delta t^3 \\ \frac{1}{2} \Delta t^2 \\ \Delta t \end{bmatrix} \dot{a}_k^e, \quad (4)$$

where $u_k^e = \dot{a}_k^e$. This allows a smooth interpolation between two behavior states with polynomials and therefore serves as a better basis for the low level optimization as shown in Section IV-B. However, the action set $\mathcal{A} = \{a^{(1)}, \dots, a^{(n)}\}$ is defined by different acceleration values in the succeeding state. This allows faster changes of the acceleration and thereby more flexible maneuvers. Consequently, \dot{a}_k^e is calculated by $\dot{a}_k^e = (a_{k+1}^e - a_k^e)/\Delta t$. As a reactive prediction model for other traffic participants, exemplary the IDM is used [13]. Accordingly, $g^o(\cdot)$ is given by the IDM in combination with the constant acceleration (CA) transition model. The IDM is defined by,

$$a_k^o = a_{\text{IDM}} \left(1 - \left(\frac{v_k}{v_{\text{des}}} \right)^\delta - \underbrace{\left(\frac{s^*(v_k, \Delta v_k)}{\Delta s_k} \right)^2}_{I_{\text{IDM},k}} \right), \quad (5)$$

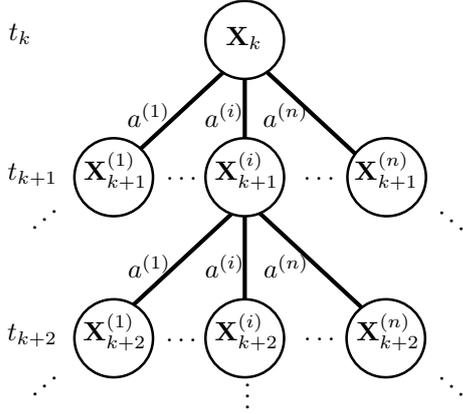


Fig. 1: Exemplary section of the utilized behavior graph. Vertices represent states connected by edges associated with actions contained in $\mathcal{A} = \{a^{(1)}, \dots, a^{(n)}\}$. The graph search is utilized to find the optimal sequence of actions up to the horizon t_τ .

where Δs denotes the distance to the leading vehicle and I_{IDM} describes the interaction term of the IDM with

$$s^*(v, \Delta v) = s_0 + vT + \frac{v\Delta v}{2\sqrt{a_{\text{IDM}}b_{\text{comf}}}}, \quad (6)$$

and $\Delta v = v_{\text{lead}} - v$, where v_{lead} is the velocity of the leading vehicle. The transition model is given by

$$\underbrace{\begin{bmatrix} s_{k+1}^o \\ v_{k+1}^o \end{bmatrix}}_{\mathbf{x}_{k+1}^o} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} s_k^o \\ v_k^o \end{bmatrix} + \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \end{bmatrix} a_k^o. \quad (7)$$

As a result, the complete state for a time step k can be described as

$$\mathbf{X}_k = [\mathbf{x}_k^e, \mathbf{x}_k^{1\dots m}], \quad (8)$$

which is a concatenation of the ego vehicle state and the states of the predicted vehicles. A schematic representation of the resulting behavior graph and according states is depicted in Fig. 1. In addition to the possible state transition, the costs for states have to be defined. Therefore, the costs for a given state \mathbf{X}_k are formulated as follows

$$J_k^e = \omega_f j_{f,k} + \omega_v j_{v,k} + \omega_{\text{jerk}} j_{\text{jerk},k} + \omega_{\text{inter}} j_{\text{inter},k}, \quad (9)$$

where $\omega = [\omega_f, \omega_v, \omega_{\text{jerk}}, \omega_{\text{inter}}]$ defines different weightings for the single cost terms. Consequently, the overall cost function is

$$J^e(\mathbf{X}, \mathbf{u}^e, \kappa(s^e), \mathbf{C}^e, g^o(\mathbf{X}, \mathbf{u}^e, \kappa(s^o), \mathbf{C}^o)) = \sum_{k=0}^{\tau} J_k^e. \quad (10)$$

By penalizing the jerk

$$j_{\text{jerk},k} = (\dot{a}_k^e)^2, \quad (11)$$

costs for varying acceleration values are induced, which leads to more stable and comfortable trajectories. The costs for the velocity deviation of the desired velocity v_{des} are

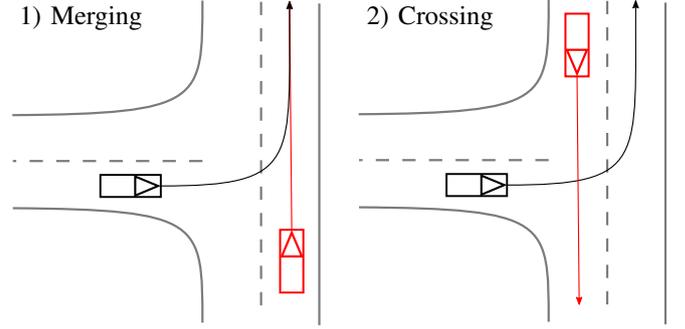


Fig. 2: Exemplary crossing and merging scenario, where the ego vehicle (black) performs a left turn without right of way. The other vehicle (red) has right of way.

chosen as follows

$$j_{v,k} = \begin{cases} (v_k^e - v_{\text{des}}(s_k^e))^2 & \text{if } v_k^e > v_{\text{des}}(s_k^e) \\ |v_k^e - v_{\text{des}}(s_k^e)| & \text{if } v_k^e \leq v_{\text{des}}(s_k^e) \end{cases}. \quad (12)$$

This cost term ensures to reach the desired velocity v_{des} and a soft constraint is utilized in order to stay below v_{des} . The costs for following other vehicles is calculated using the interaction term of the IDM

$$j_{i,k} = I_{\text{IDM},k}. \quad (13)$$

This allows the dynamic consideration of speed and spatial difference to the leading vehicle. As a result, an appropriate gap to the vehicle in front is kept. Following the definition of the courtesy term in [6], the interaction costs for other traffic participants are formulated as the costs arising due to actions of the ego vehicle. Therefore, induced costs for other vehicles are considered by

$$j_{\text{inter},k} = \sum_o |a_{\text{norm},k}^o - a_{\text{inter},k}^o|, \quad (14)$$

where $a_{\text{norm},k}^o$ is the predicted acceleration for any other vehicle if the ego vehicle is without influence. In contrast, $a_{\text{inter},k}^o$ is the predicted acceleration if the ego vehicle interacts with the other vehicle. In this work, this generic concept is exemplarily modeled for two different interaction types at an intersection where the other vehicle has right of way. The scenarios are also depicted in Fig. 2:

1) *Merging Scenario*: A merging scenario means that the ego vehicle's and the other vehicle's center lines are merging to the same center line. If the ego vehicle pulls out in front of the other vehicle, costs for the other vehicle are induced. This is because the other vehicle might have to decelerate in order to keep a safety gap to the ego vehicle. Therefore, $a_{\text{inter},k}^o$ is modeled with Equation 5 by using the ego vehicle as leader. The other vehicle is assumed to react to the ego vehicle as soon as the ego vehicle enters the intersection. In contrast, $a_{\text{norm},k}^o$ is calculated without the ego vehicle as if the ego vehicle stops at the intersection.

2) *Crossing Scenario*: This scenario covers situations in which the ego vehicle crosses the lane of another vehicle. Therefore, a spatio-temporal constraint c^e is derived in which

an intersection zone with a certain safety distance is not allowed to be occupied by the ego vehicle. Using these constraints, no interaction is assumed, i.e., the other vehicle does not have to react to the ego vehicle as the ego vehicle leaves the intersection zone early enough or does not enter it before the other vehicle crosses the intersection. Following this argumentation, $a_{\text{norm}}^o = a_{\text{inter}}^o$. Thus, $j_{\text{inter}} = \infty$ if any constraint is violated. This also applies for constraints which do not emerge from crossing scenarios.

In summary, courtesy and interactions are already considered during the actual behavior planning, instead of after the trajectory generation as in [5]. In order to solve the presented problem, the A* graph search algorithm is employed [7], [14], [15]. To reduce the computational effort a heuristic function can be utilized. As presented in previous work, a reasonable approach is to exploit inevitable collision states [7]. In the context of the presented concept this means states which definitely lead to a violation of a spatio-temporal constraint c^e own heuristic costs equal to the actual costs of this violation. For more details refer to [7]. As a result of the graph search, \mathcal{T}_B is obtained and can be used for further optimization.

B. Trajectory Generation

The rough discretized behavior trajectory \mathcal{T}_B is used to generate a smooth and continuous execution trajectory \mathcal{T}_{ex} . In contrast to [1] and [16], the trajectories are represented by piecewise septic polynomials with fixed temporal spacing and multiple target states for different times are regarded. In addition to acceleration continuity, septic polynomials allow for continuous jerk [17]. Therefore, the trajectory fulfills

$$\mathcal{T}_{\text{ex},i}^{(n)}(t_{i+1}) \stackrel{!}{=} \mathcal{T}_{\text{ex},i+1}^{(n)}(t_{i+1}) \quad \text{and} \quad n \in [0, 3], \quad (15)$$

where $\mathcal{T}_{\text{ex},i}^{(n)}$ is the n^{th} derivation of the polynomial which defines \mathcal{T}_{ex} from t_i to t_{i+1} . In general, septic polynomials are defined by their coefficients $c_{0\dots7}$ with

$$\mathbf{x}(t) = \underbrace{\begin{bmatrix} 1 & t & t^2 & t^3 \\ 0 & 1 & 2t & 3t^2 \\ 0 & 0 & 2 & 6t \\ 0 & 0 & 0 & 6 \end{bmatrix}}_{\mathbf{M}_1(t)} \mathbf{c}_{0123} + \underbrace{\begin{bmatrix} t^4 & t^5 & t^6 & t^7 \\ 4t^3 & 5t^4 & 6t^5 & 7t^6 \\ 12t^2 & 20t^3 & 30t^4 & 42t^5 \\ 24t & 60t^2 & 120t^3 & 210t^4 \end{bmatrix}}_{\mathbf{M}_2(t)} \mathbf{c}_{4567}, \quad (16)$$

where $\mathbf{x}(t) = [s, v, a, \dot{a}]^T$. Furthermore, the coefficients can be derived from a given start state \mathbf{x}_0 , terminal state \mathbf{x}_f and an according time difference t_f . As a result, $c_{0\dots7}$ can be determined by

$$\mathbf{c}_{0123} = \mathbf{M}_1^{-1}(0)\mathbf{x}_0, \quad (17)$$

$$\mathbf{c}_{4567} = \mathbf{M}_2^{-1}(t_f)(\mathbf{x}(t_f) - \mathbf{M}_1(t_f)\mathbf{c}_{0123}). \quad (18)$$

For this reason, septic polynomials can be used to interpolate behavior states \mathbf{x}_k^e and \mathbf{x}_{k+1}^e . In contrast to [7] the utilized transition model described with Equation 4 enables smooth interpolations as shown in Fig. 3. However, there might be more comfortable solutions so that not only immediately succeeding states are interpolated. With Algorithm 1 multiple

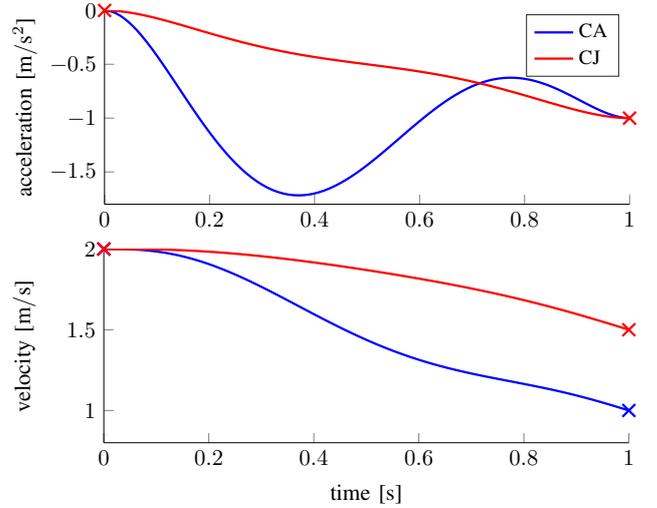


Fig. 3: Exemplary comparison of directly interpolated states with the constant acceleration (CA) transition model used in the state of the art [7] and the presented constant jerk (CJ) approach. Interpolation is done with septic polynomials, where the start state is defined by $x_0 = [0 \text{ m}, 2 \text{ m s}^{-1}, 0 \text{ m s}^{-2}]^T$ and the action by $a_0 = -1 \text{ m s}^{-2}$. With CA this leads to a target state $x_1 = [1.5 \text{ m}, 1 \text{ m s}^{-1}, -1 \text{ m s}^{-2}]^T$. Using CJ results in the target state $x_1 = [1.833 \text{ m}, 1.5 \text{ m s}^{-1}, -1 \text{ m s}^{-2}]^T$. For comparability $\dot{a}_0 = 0 \text{ m s}^{-3}$ and $\dot{a}_1 = 0 \text{ m s}^{-3}$ is set for both scenarios.

trajectory candidates are generated. In general, the start state \mathbf{x}_s^e is interpolated to each following state \mathbf{x}_k^e . The rest of the states from \mathbf{x}_k^e to \mathbf{x}_τ^e can be directly interpolated. More sophisticated solutions can be found by using the algorithm recursively in a way that each state \mathbf{x}_k^e is again taken as start state. However, with the number of recursions l_r the number of trajectory candidates which have to be evaluated increases. In Fig. 4, some exemplary trajectory candidates are shown with according states of \mathcal{T}_B .

Algorithm 1 Generation of trajectory candidates

```

get start state  $\mathbf{x}_s^e$  from  $\mathcal{T}_B$ 
 $k \leftarrow s + 1$ 
while  $k \neq \tau$  do
    traj  $\leftarrow$  interpolate( $\mathbf{x}_s^e, \mathbf{x}_k^e$ )
     $j \leftarrow k$ 
    while  $j \neq \tau$  do
        traj.append(interpolate( $\mathbf{x}_j^e, \mathbf{x}_{j+1}^e$ ))
         $j \leftarrow j + 1$ 
    end while
     $k \leftarrow k + 1$ 
    traj_set.add(traj)
end while

```

After the generation of trajectory candidates, these can be verified against the constraints \mathbf{C}^e , kinematic boundaries and

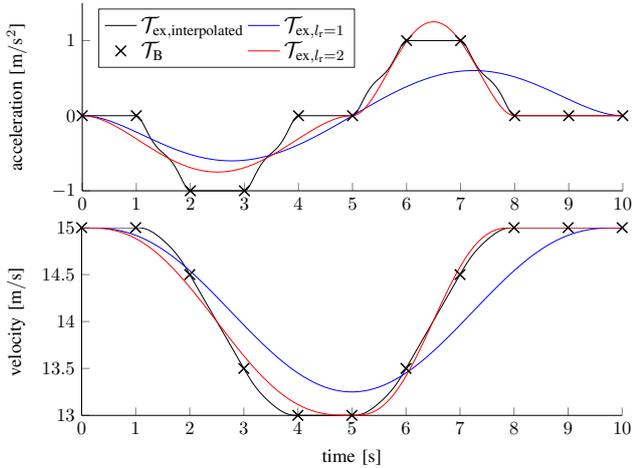


Fig. 4: Exemplary trajectory candidates for given behavior states. In black all states are directly interpolated. In blue the interpolation between first and last state. In red an example for $l_r = 2$, where the polynomials reach from $k = 0$ to $k = 5$ to $k = 8$ and the rest of the states are directly interpolated.

evaluated with a cost functional optimizing the comfort

$$J_{ex} = \int_{t_0}^{t_\tau} \dot{a}_{ex}^2 dt. \quad (19)$$

Furthermore, there are additional constraints introduced in order to omit velocity overshooting, with

$$v_{min}^e \leq v_{ex}(t) \leq v_{max}^e \quad t \in [t_0, t_\tau], \quad (20)$$

where v_{min}^e and v_{max}^e represents the minimal and maximal velocity value contained in \mathcal{T}_B . Finally, the best valid trajectory can be chosen.

In contrast to state of the art concepts, the presented concept considers the whole horizon t_τ for the low level optimization of \mathcal{T}_B [7]. Therefore, constraints can be checked for the whole horizon and consequently safety is increased. In addition, the sampling effort can be drastically reduced as the behavior states can be smoothly interpolated and it is not necessary to sample states which deviate from \mathcal{T}_B . As a result, smooth and safe trajectories over the whole horizon are received.

C. Replanning

As described in Section IV-B, \mathcal{T}_{ex} might deviate from \mathcal{T}_B . In order to keep consistency, the start state used for behavior replanning is not the actual position $\mathbf{x}_{ex}(t)$ contained in \mathcal{T}_{ex} , but rather the predicted state on the behavior trajectory $\mathbf{x}^e(t)$. As a result, in static environments the behavior trajectory is consistent for subsequent planning steps. However, there is a huge impact on the solutions if the environment changes, as there might exist no smooth interpolation to the new next behavior state. For this reason, in this work an additional behavior trajectory is calculated beginning with $\mathbf{x}_{ex}(t)$. In this behavior trajectory a smooth interpolation of the current state to the next behavior state is guaranteed and consequently smooth reactions to changing predictions

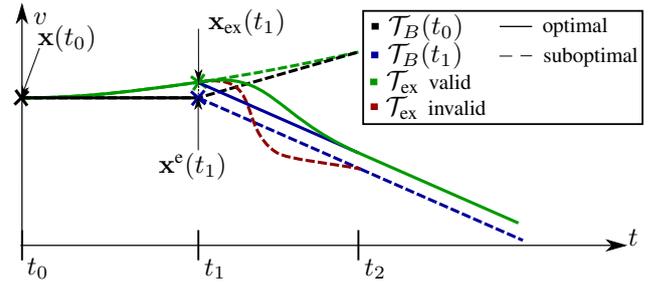


Fig. 5: Schematic illustration of replanning strategy with changing environment in t_1 . The start state is denoted by $\mathbf{x}(t_0)$, the actual position in t_1 by $\mathbf{x}_{ex}(t_1)$ and the according position on the behavior trajectory by $\mathbf{x}^e(t_1)$. Dashed lines are planned states which are suboptimal and not used for driving. The behavior trajectory planned in t_0 is shown in black the ones planned in t_1 are displayed in blue. Red is the non-smooth or even invalid interpolation using the behavior trajectory starting at $\mathbf{x}^e(t_1)$. Consequently, the behavior trajectory beginning from $\mathbf{x}_{ex}(t_1)$ with its associated interpolation is used for the driven trajectory which is depicted in green.

and environment. In the end, trajectory candidates for both behavior trajectories can be generated and the best candidate can be chosen. The according problem is illustrated in Fig. 5. Furthermore, as both behavior trajectories can be calculated in parallel, the additional effort should only have minor influence on the runtime.

V. IMPLEMENTATION AND EVALUATION

For the evaluation, an intersection in Ulm-Lehr Germany is investigated and the according center lines are taken from a high-precision digital map of Ulm University [4]. A corresponding overview is depicted in Fig. 6a, where the visualization is done in *CoInCar-Sim* [18]. The ego vehicle (ID 1) approaches the intersection from the left and has to perform a left turn. However, vehicle 2 (ID 2) has right of way and is approaching from the right. In one scenario, vehicle 3 (ID 3) is regarded which is approaching from top with right of way and crosses the line of the ego vehicle by driving straight. In general, three different scenarios are regarded. In scenario 1 only the ego vehicle and vehicle 2 are involved. The ego vehicle is parametrized with $\omega_{inter}^1 = 20$ which means that induced costs for other vehicles are less penalized. As a result, more aggressive behavior is expected. Scenario 2 is equal to scenario 1 except that $\omega_{inter}^h = 50$ is used. These high interaction costs will lead to more courteous behavior. In scenario 3 all vehicles are regarded and ω_{inter}^1 is used. Thereby, vehicle 3 crosses the ego route before vehicle 2 passes the intersection. An overview of the scenarios with the associated interaction cost weighting and involved vehicles can be found in Table I. In order to emphasize the practicability of the approach and verify the robustness against uncertainties, the other vehicles are simulated with CV in contrast to the prediction model used. This means the other vehicles do not choose a

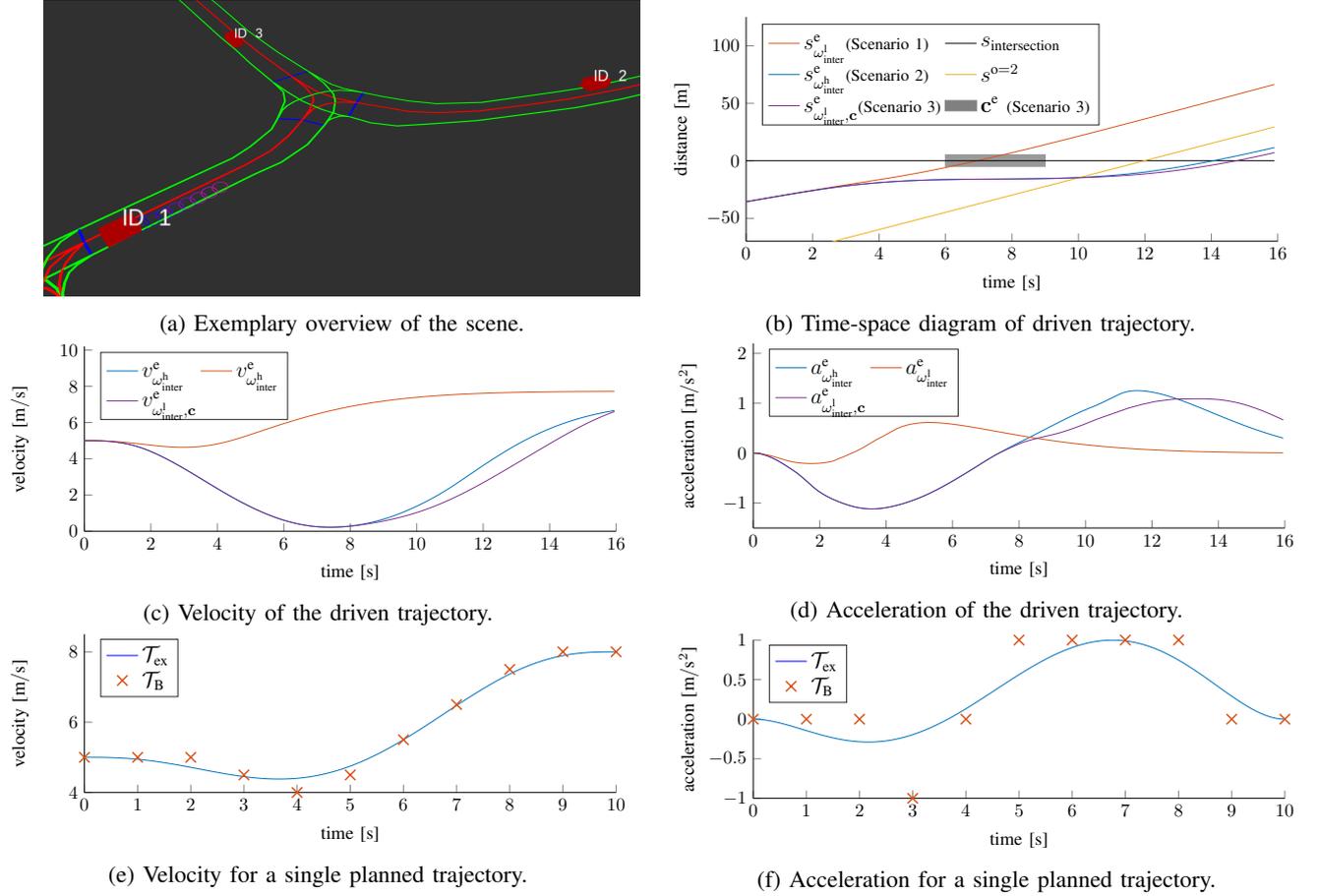


Fig. 6: Evaluation Scenario. (a): Exemplary overview of the scene visualized *CoInCar-Sim* [18]. Lanes are represented by lane boundaries, the ego vehicle has ID 1 and the other vehicles ID 2 and 3. (b)-(d): Illustration of the driven trajectories. Scenario 1 and 2 do not include vehicle 3 and show merging behavior for different ω_{inter} , where vehicle 2 is drawn in yellow. In scenario 3, vehicle 3 manifests through spatio-temporal constraint shown with a gray rectangle. (e)-(f): Planned trajectory at the beginning of scenario 1, including behavior states and interpolated results.

decelerating reaction but rather drive aggressive. However, the ego vehicle is expected to enforce a courteous maneuver and keep an appropriate time gap to the other vehicles due to continuous replanning. The concept is implemented in C++ using the A*-implementation of the Discrete Optimal Search Library (DOSL) [15]. The according runtime is evaluated on a i5-7300U CPU with 2.6 GHz. For behavior planning the action set $\mathcal{A} = [-2, -1, 0, 1, 2] \text{m/s}^2$ is chosen, where the maximum acceleration difference between two subsequent behavior states is set to $\Delta a = 1.9 \text{m/s}^2$ in order to restrict the longitudinal jerk. Replanning is done with a frequency of 5Hz. Further parameters are presented in Table II.

TABLE I: Scenario Overview. It is shown which vehicles are regarded and which interaction cost weighting is used.

Scenario	ω_{inter}^l	ω_{inter}^h	Vehicle 2	Vehicle 3
Scenario 1	✓	✗	✓	✗
Scenario 2	✗	✓	✓	✗
Scenario 3	✓	✗	✓	✓

TABLE II: Parameters used for evaluation

ω_{inter}^l	20	t_τ	10 s	T	1.5 s
ω_{inter}^h	50	a_{max}	$2.5 \frac{\text{m}}{\text{s}^2}$	δ	4
ω_f	5	a_{min}	$-2.5 \frac{\text{m}}{\text{s}^2}$	v_{des}	$7.5 \frac{\text{m}}{\text{s}}$
ω_v	1	v_{max}	$10 \frac{\text{m}}{\text{s}}$	a_{IDM}	$0.73 \frac{\text{m}}{\text{s}^2}$
ω_{jerk}	1	l_r	1	b_{comf}	$1.67 \frac{\text{m}}{\text{s}^2}$
		Δt	1 s	s_0	2 m

First, a single planning step is considered in Fig. 6e and 6f. Thereby, the trajectory for the first planning step in scenario 1 is shown. In the beginning of the horizon, the speed has to be reduced in order to limit lateral acceleration due to high curvature of the center line. Subsequently, the velocity increases drastically in order to reduce the interaction costs caused by the upcoming vehicle 2. By sampling quintic polynomials with manual heuristics as done in previous work, such generic behavior is hard to achieve [5]. Because on the one hand, only one target state can be set and therefore multiple objectives are difficult to enforce,

as e.g. low speed at a specific longitudinal position and fast acceleration after crossing the intersection. On the other hand, the form of a single quintic polynomial might not even allow the desired behavior, as e.g. multiple subsequent acceleration and deceleration periods. In addition to the first planning step, the driven trajectories for different scenarios are displayed in Fig. 6b-6d. In scenario 1, ω_{inter}^l is used which means the induced costs are less penalized. Consequently, the ego vehicle merges into the lane in front of vehicle 2 which has right of way. However, the behavior can still be considered courteous since there is sufficient space to vehicle 2 due to a low speed reduction and fast acceleration after the crossing. In the second scenario, induced costs are heavily penalized with ω_{inter}^h . This leads to even more courteous behavior and the ego vehicle lets the other vehicle pass the intersection first and then follows with an appropriate safety gap. In scenario 3, the ego vehicle is again parametrized with ω_{inter}^l . In contrast to scenario 1, the ego vehicle has to wait for the crossing vehicle 3. As a result, the ego vehicle can not merge in front of vehicle 2 anymore since this would induce too much interaction costs even though the lower cost weighting is used. These results show that courteous driving and induced costs for other vehicles can be effectively considered in the presented framework.

TABLE III: Runtime evaluation, where maximum calculation time rounded to ms is shown and dummy vehicles are added to the state space of the single scenarios.

Number of vehicles	1	2	3	4	5
Scenario 1	8	8	7	9	9
Scenario 2	17	19	18	20	28
Scenario 3	n/a	10	15	13	12

The mean calculation time for the scenarios is 2.36 ms and the maximum calculation time is 17 ms. Further runtime evaluations shown in table III indicate that the size of the state space has only minor influence on the runtime. Following the argumentation in Section IV-C the runtime can even be further reduced by parallelization. Furthermore, smooth trajectories are generated even though the other vehicles do not behave as predicted in the framework with the IDM but rather with CV. This additionally emphasizes that prediction inaccuracies can be overcome by continuous replanning and the robustness of the approach.

VI. CONCLUSION

In this work, a novel, real time capable behavior and trajectory planning concept is presented. Induced costs for other traffic participants are considered by a game-theoretic approach and a courteous driving strategy is generated. The formulated optimal control problem is solved using a discrete graph-based approach including a transition model that allows for smooth interpolation between the states. Further, a trajectory generation concept based on septuple polynomials is shown that enables the generation of smooth

and safe trajectories on large horizons. Future work will include measurement and prediction uncertainties, e.g., as shown in [9]. Additionally, the concept will be extended to other traffic use cases and implemented on the experimental vehicle of Ulm University.

REFERENCES

- [1] J. Villagra, V. Milans, J. Prez, and J. Godoy, "Smooth path and speed planning for an automated public transport vehicle," *Robotics and Autonomous Systems*, vol. 60, no. 2, pp. 252 – 265, 2012. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S092188901100203X>
- [2] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, "Optimal trajectory generation for dynamic street scenarios in a frenet frame," in *2010 IEEE International Conference on Robotics and Automation*, May 2010, pp. 987–993.
- [3] J. Ziegler, P. Bender, T. Dang, and C. Stiller, "Trajectory planning for bertha 2014; a local, continuous method," in *2014 IEEE Intelligent Vehicles Symposium Proceedings*, June 2014, pp. 450–457.
- [4] F. Kunz, D. Nuss, J. Wiest, H. Deusch, S. Reuter, F. Gritschneider, A. Scheel, M. Stbler, M. Bach, P. Hatzelmann, C. Wild, and K. Dietmayer, "Autonomous driving at ulm university: A modular, robust, and sensor-independent fusion approach," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, June 2015, pp. 666–673.
- [5] N. Evestedt, E. Ward, J. Folkesson, and D. Axehill, "Interaction aware trajectory planning for merge scenarios in congested traffic situations," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Nov 2016, pp. 465–472.
- [6] L. Sun, W. Zhan, M. Tomizuka, and A. D. Dragan, "Courteous autonomous cars," *arXiv preprint arXiv:1808.02633*, 2018.
- [7] C. Hubmann, M. Aeberhard, and C. Stiller, "A generic driving strategy for urban environments," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Nov 2016, pp. 1010–1016.
- [8] J. Ziegler and C. Stiller, "Spatiotemporal state lattices for fast trajectory planning in dynamic on-road driving scenarios," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2009, pp. 1879–1884.
- [9] E. Ward and J. Folkesson, "Towards risk minimizing trajectory planning in on-road scenarios," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, June 2018, pp. 490–497.
- [10] S. Brechtel, T. Gindele, and R. Dillmann, "Probabilistic decision-making under uncertainty for autonomous driving using continuous pomdps," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, Oct 2014, pp. 392–399.
- [11] 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*, vol. 3, no. 1, pp. 5–17, March 2018.
- [12] H. Bai, S. Cai, N. Ye, D. Hsu, and W. S. Lee, "Intention-aware online pomdp planning for autonomous driving in a crowd," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 454–460.
- [13] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical review E*, vol. 62, no. 2, p. 1805, 2000.
- [14] S. J. Russell and P. Norvig, *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited., 2016.
- [15] S. Bhattacharya, "Discrete optimal search library (dosl): A template-based c++ library for discrete optimal search," 2017. [Online]. Available: <https://github.com/subh83/DOSL>
- [16] S. Liu, "An on-line reference-trajectory generator for smooth motion of impulse-controlled industrial manipulators," in *7th International Workshop on Advanced Motion Control. Proceedings (Cat. No. 02TH8623)*, July 2002, pp. 365–370.
- [17] C. Rathgeber, F. Winkler, X. Kang, and S. Müller, "Optimal trajectories for highly automated driving," *World Academy of Science, Engineering and Technology, International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering*, vol. 9, no. 6, pp. 969–975, 2015.
- [18] M. Naumann, F. Poggenhans, M. Lauer, and C. Stiller, "Coincar-sim: An open-source simulation framework for cooperatively interacting automobiles," in *IEEE Intl. Conf. Intelligent Vehicles*, 2018.