

Trajectory Planning for Automated Vehicles using Driver Models

Maximilian Graf¹

Oliver Speidel¹

Julius Ziegler²

Klaus Dietmayer¹

Abstract—Behavioral-specific trajectory planning for automated vehicles is an intensively explored research topic. Many situations in daily traffic, e.g. following a leading vehicle or stopping behind it, require knowledge about how the scene may evolve. In recent years, much effort has been put into developing driver models to predict traffic scenes as realistic as possible according to human behavior. In this paper, we present a method for behavioral-specific trajectory planning using dedicated driver models. The main idea is to first calculate a reference trajectory using a suitable model to achieve the desired behavior and then to incorporate this reference trajectory into an optimal control problem to obtain an acceleration- and jerk-optimal trajectory. A major strength of this method is in the small computation time, since the problem is formalized as a quadratic optimization problem and can thus be efficiently solved in real time, even for a huge number of optimization variables.

I. INTRODUCTION

A. Motivation

Trajectory planning is a basic requirement to autonomous driving and must be present in every automatically controlled vehicle. Consequently much effort has been put into this field and several techniques have been developed already [1], [2]. In general it can be distinguished between sampling based and continuous approaches. The former discretize the search space by generating a finite set of candidate trajectories before selecting the "best" one, i.e. one that satisfies the constraints and optimizes the cost functional. Sampling based approaches are typically used in complex and unstructured environments, where a cost function contains many local minima and an initial guess may be hard to find. Since the search space is discrete, the trajectory can only be optimal up to the resolution limit of the discretization. If the cost functional contains higher derivatives of the trajectory, it is difficult to systematically discretize the search space in a way that really approaches the continuum of all trajectories. The methods commonly used are of heuristical nature. In contrast, continuous methods use optimal control techniques to optimize an initial guess for a trajectory in terms of a cost function while respecting constraints. In well structured environments, where a decent initial guess can be determined and the problem is well conditioned, convergence to a desired local optimum can be achieved reliably. This increases the quality of the optimal solution in contrast to sampling

based approaches. Thus, the environmental structure plays an important role for the performance of trajectory planning algorithms and should be exploited if possible. In [3] a local continuous approach in a well structured environment is presented. This method was used for the Bertha Benz drive in 2013 [4] and performed well on the route of more than 65 miles from Mannheim to Pforzheim (Germany). The cost functional in [3] is designed in a way, that the resulting trajectory keeps the vehicle in the middle of the lane, by approaching the target speed (which is usually the speed limit). The cost functional and constraints used in [3] are tailored to the case of on-road cruising and the evasion of static obstacles that narrows down the driving corridor. Other behavior, such as car following or stopping is achieved heuristically by a conventional adaptive cruise controller that overrides the longitudinal commands to the vehicle controller if necessary. Hence, the optimal planning scheme does not always actually affect the longitudinal control input. In this paper we present an approach, that generates a behavioral-specific reference trajectory, that is incorporated into a quadratic optimal control problem (OCP) to obtain an acceleration- and jerk-optimal trajectory for achieving comfortable driving behavior. Subsequently, we will use the word "smooth" to indicate trajectories that are optimized in terms of acceleration and jerk. Furthermore, it is worth mentioning, that due to the quadratic nature, convergence can be achieved fast. This guarantees not only real-time capability of this method even in case of a large number of optimization variables but makes the approach also extendable.

The remainder of the paper is as follows: In Section I-B we give an overview of related work, show common trajectory planning algorithms as well as approaches, closely related to ours. The theoretically part of this paper's contribution is described in section II. Therein the use of driver models for our approach as well as the optimal control problem are elaborated in more detail. In section III we show the results of our method in different driving scenarios.

B. Related Work

A popular approach for sampling based trajectory planning is presented in [5] where jerk-optimal polynomials are sampled in a Frenét frame. Combining every longitudinal with every lateral polynomial leads to a set of trajectories, from which the one, that satisfies constraints while being optimal in terms of a given cost function is chosen. This method was successfully applied in several autonomous driving applications. Nevertheless, because of the short temporal horizon,

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

² Julius Ziegler is with Atltec GmbH, 76131 Karlsruhe, Germany, e-mail: ziegler@atltec.de.

the method has only a very low foresight. Also trajectories are restricted due to the polynomial nature of the samples. Another well-known approach is presented in [6]. Therein a search tree consisting of path primitives is utilized. Temporal information is generated by sampling velocity profiles over the paths. A search graph algorithm is used to find the best combination of paths and velocity profiles to select the best collision free trajectory according to a cost function. To get a more foresighted behavior, the temporal horizon can be increased by investigating the search tree for more paths and samples of velocity profiles. This puts an heavy computational burden on the method.

If the environment is structured, continuous methods can be utilized to overcome the disadvantages of sampling based approaches. An example can be found in [3]. Therein an initial guess is optimized utilizing methods from continuous optimization theory. The trajectory can be optimized over a relatively large horizon, by maintaining an appropriate computing time. The aspect of foresight, together with the good condition of the optimization problem make the approach in [3] very attractive. Since this paper focuses on behavioral-specific trajectory planning, we also want to present some preliminary work in this field.

In [7] a concept is described, in which behavioral-specific spatio-temporal corridors are computed from a grid laid out over the road. These corridors are then used to constrain the search space and thus enforce the desired behavior. A similar concept is shown in [8], where a first rough trajectory is estimated by using A*-search on a discretized action set. The resulting strategy is then realized by using the trajectory planning algorithm in [5]. In [9] it is described, how overtaking maneuvers based on the concept described in [3] can be realized. Therefore maneuvers are enforced by constraints, without changing the cost function. The concept presented in [10] enforces the behavioral strategy by using a risk-averse cost map which is based on the prediction of the traffic situation. The according trajectory is then calculated by utilizing the RRT* algorithm. However, the concept suffers of a high computational effort which prohibits its employment in real time applications. An opportunity for cooperative motion planning in a platoon of cars is described in [11]. Therefore, only the longitudinal trajectory is optimized by solving an optimal control problem based on the concept in [3]. Distance keeping to the leading vehicle is achieved by adding an extra term to the optimization problem to penalize small distances. Calculating trajectories for lane keeping is a well-investigated topic and already integrated in diverse autonomous driving test carriers or even plugged in productive cars. An opportunity, to realize complex maneuvers is to integrate social interaction aware models into trajectory planning modules. None of the presented methods perform trajectory planning directly by using interaction aware driver models in real time. We want to show an opportunity of how to use such models for real time trajectory planning. In a first approach, we use the well-known "intelligent driver model" (IDM) [12] to show that the method works. The method is easily extendable

for more complex models like e.g. [13].

II. TRAJECTORY PLANNING

A. Driver Model

The main idea of our approach is to efficiently integrate driver models into trajectory planning. In general arbitrary models, that allow the prediction of trajectories can be used. In a first approach, we use the IDM to plan trajectories for driving within a single lane (driving corridor). All other vehicles are assumed to move within the same driving corridor and in the same principal direction.

Differential equations of the IDM are as follows:

$$\dot{v}_\alpha = a^{(\alpha)} \left[1 - \left(\frac{v_\alpha}{v_0^{(\alpha)}} \right)^\delta - \left(\frac{s^*(v_\alpha, \Delta v_\alpha)}{\Delta s_\alpha} \right) \right]. \quad (1)$$

The desired minimum gap s^* is given by

$$s^*(v, \Delta v) = s_0^{(\alpha)} + T^{(\alpha)}v + \frac{v\Delta v}{2\sqrt{a^{(\alpha)}b^{(\alpha)}}}. \quad (2)$$

Where v_0 is the desired velocity (often considered as the speed limit), T the safe time headway, a the maximum acceleration, b the comfortable deceleration, δ the acceleration exponent and s_0 the standstill distance. The index α identifies the individual units of the convoy of vehicles. The distance to the leading vehicle (indicated by index $\alpha - 1$) is referred to as $\Delta s_\alpha = s_{\alpha-1} - s_\alpha - l$, with the vehicle length l . The relative velocity is indicated as Δv_α . As becomes apparent, the equations describe the vehicles acceleration depending on its longitudinal state, the leading vehicle as well as several parameters, that have to be carefully identified to reproduce human like behavior. Equation 1 can be split into two terms: The free-cruising term $1 - (v_\alpha/v_0^{(\alpha)})^\delta$ and the interaction term $s^*(v_\alpha, \Delta v_\alpha)/s_\alpha$. The free cruising term converges to zero for velocities going against v_0 . The interaction term decreases if the desired distance s^* gets small and even vanishes for large distances Δs_α to the leading vehicle. The IDM can of course also be used if there are no further traffic participants on the street. In this case, the interaction term is omitted.

Since equations 1 and 2 merely model longitudinal behavior, an extension has been made in [14] to also consider the road's curvature. The main idea is to limit the lateral acceleration a_{lat} due to the velocity. Therefore, the curvature $\kappa(s)$ of the line, to which the IDM is referred to, is calculated over the longitudinal coordinate s of the Frenét frame. In a second step, the maximum allowed velocity is calculated by

$$v_{max}(s) = \min \left(\sqrt{\frac{a_{lat}}{\kappa(s)}}, v_{lim} \right). \quad (3)$$

With the speed limit indicated by v_{lim} . Subsequently, the velocity gradient dv_{max}/ds must be limited to decrease the velocity already before sections of high curvature are reached to obtain a more foresighted behavior.

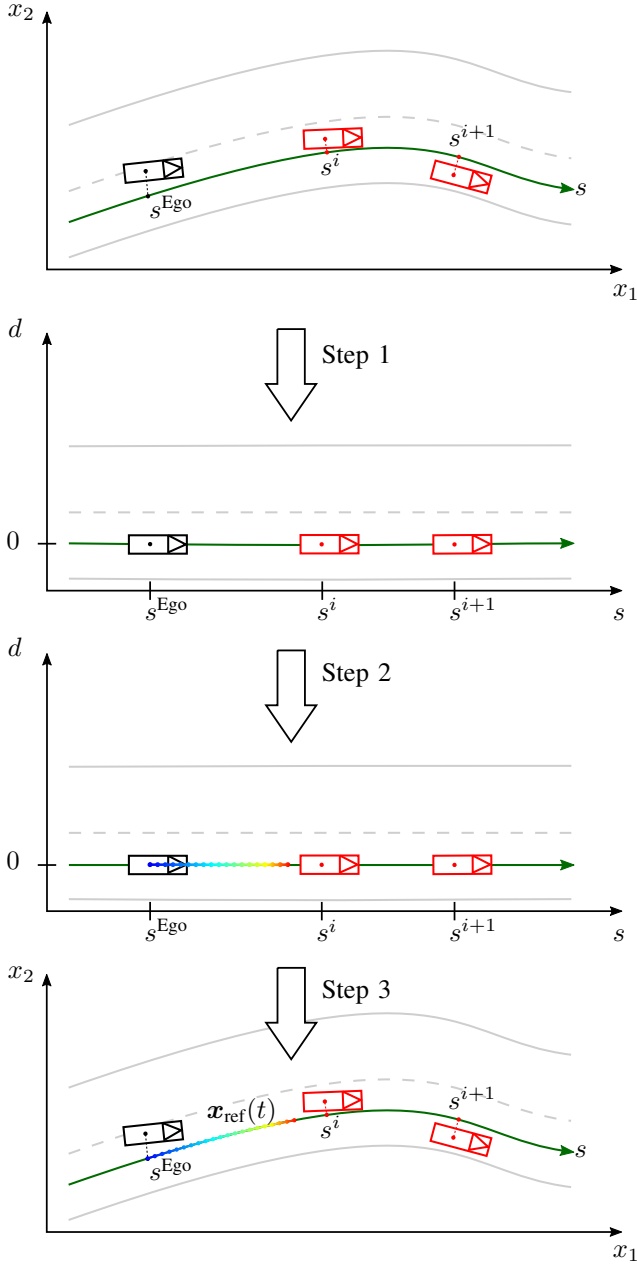


Fig. 1. Calculation of the reference trajectory. Step 1: Vehicle states are transformed into Frenét coordinates of the centerline. Step 2: IDM prediction is done for each traffic participant. Step 3: Only the predicted trajectory of the ego vehicle is transformed back into cartesian coordinates and used as the reference trajectory for the subsequent trajectory optimization.

B. Trajectory generation

The environment model consists of the lanes centerline, the speed limit, the position, orientation, velocity and acceleration of surrounding vehicles as well as of the ego vehicle. Planning takes place in a 2D cartesian coordinate system, where the centerline is represented by a 2D polygonal line and reference point positions of vehicles can be described as 2D points. For further calculations, the temporal planning interval is discretized by a number of N support points $[t_0, t_1, \dots, t_{N-1}]$. Position states of the trajectory at time t_i

are referred as $\mathbf{x}(t_i) = \mathbf{x}_i$, with $i = 0, \dots, N - 1$. The general procedure is then as follows: In a first step, each traffic participant has to be transformed into the Frenét frame of the centerline. Therefore, the method described in [3] is utilized (step 1).

The points on the centerline are the projected positions and mark the corresponding s coordinates of the vehicles. If the orientation difference between a vehicle and the center line at the point of projection is small (which is usually the case), velocities can be maintained in Frenét coordinates. After transformation, the whole platoon of vehicles, including the ego vehicle is predicted over the planning horizon T using IDM equations 1 and 2 (step 2). The predicted trajectory of the ego vehicle must then be transformed back into cartesian coordinates (step 3). The individual steps are shown in figure 1. The color coding of the reference trajectory $\mathbf{x}_{\text{ref}}(t)$ refers to the spatio-temporal relation between the trajectory support points, where blue depicts the beginning of the temporal planning interval and red its end respectively.

There are three problems with the reference trajectory $\mathbf{x}_{\text{ref}}(t)$: First, the path is directly on the centerline, which is not necessarily smooth. Second, the velocity calculated by IDM equations 1 and 2 using the prior calculated velocity profile may also take non-smooth values. And finally, the initial sample of the trajectory is not identical to the ego vehicle state, hence it does not form a smooth continuation of the past trajectory. A final optimization step remedies these problems and asserts both a sufficiently smooth trajectory and a proper continuation of the past trajectory. However, $\mathbf{x}_{\text{ref}}(t)$ represents human behavior to an appropriate accuracy for our purposes and is sufficient for the subsequent procedure. The reference $\mathbf{x}_{\text{ref}}(t)$ is only used to guide the subsequent trajectory optimization.

In the next step, the final trajectory $\mathbf{x}(t)$ is calculated. Therefore it must be ensured, that $\mathbf{x}(t)$ is smooth on the one hand and on the other hand, the trajectory's initial state must be equal to the ego vehicles state. As illustrated in the last picture of figure 1, this is not the case for $\mathbf{x}_{\text{ref}}(t)$. To achieve these goals, we state a cost function based on the work in [3], as follows:

$$J[\mathbf{x}(t)] = \int_0^T L(\mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}}) dt. \quad (4)$$

With

$$L = w_{\text{spatial}} j_{\text{spatial}} + w_{\text{acc}} j_{\text{acc}} + w_{\text{jerk}} j_{\text{jerk}}, \quad (5)$$

where

$$j_{\text{spatial}} = \|\mathbf{x}(t) - \mathbf{x}_{\text{ref}}(t)\|_2^2. \quad (6)$$

The spatial term penalizes the deviation between the trajectory and the reference trajectory. If there was only a spatial term, the OCPs solution would be identical to the reference trajectory.

To obtain smooth behavior, acceleration as well as jerk are penalized by

$$j_{\text{acc}} = \|\ddot{\mathbf{x}}(t)\|_2^2 \quad (7)$$

TABLE I
PARAMETERS USED FOR IDM

v_0	T	a	b	δ	s_0	l
13.66m s^{-1}	2	2	2	4	2	5

and

$$j_{\text{jerk}} = \|\ddot{\mathbf{x}}(t)\|_2^2. \quad (8)$$

The weights w_{spatial} , w_{acc} and w_{jerk} are used to penalize the terms individually. Since our trajectory is represented as a discrete set of N 2D points, derivatives have to be approximated. This is done as follows:

$$\ddot{\mathbf{x}}_i \approx \mathbf{x}_{\text{dd},i} = \frac{\mathbf{x}_{i+1} - 2\mathbf{x}_i + \mathbf{x}_{i-1}}{\Delta t^2}, \quad (9)$$

for the acceleration and

$$\ddot{\mathbf{x}}_i \approx \mathbf{x}_{\text{ddd},i} = \frac{-\mathbf{x}_{i-2} + 3\mathbf{x}_{i-1} - 3\mathbf{x}_i + \mathbf{x}_{i+1}}{\Delta t^3} \quad (10)$$

for the jerk. Thus, the cost functional is approximated by a sum:

$$J(\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_{N-1}) = \sum_{i=2}^{N-2} L(\mathbf{x}_i, \mathbf{x}_{\text{dd},i}, \mathbf{x}_{\text{ddd},i}) \Delta t + w_{\text{spatial}} \|\mathbf{x}_{\text{ref},N-1} - \mathbf{x}_{N-1}\|_2^2 \Delta t. \quad (11)$$

The multiplication by the constant parameter Δt can be neglected, since it has no influence on the optimal solution. To enforce a smooth continuation with the past trajectory, $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2$ are set to constant values instead of being free parameters of the optimization problem. This enforces initial values for the trajectory and its first two derivatives and hence, continuous orientation and curvature of the trajectory. Since equation 11 contains only quadratic terms, the OCP is unconstrained quadratic and thus closed-form solvable.

III. EVALUATION

The effectiveness of our approach is shown in two simulated scenarios. To give a qualitative impression of the computed outputs, we show a planned example trajectory on a winding road. The second scenario emphasizes longitudinal planning. The setting contains two elements that influence longitudinal steering commands: A leading vehicle and a (red) traffic light. The traffic light forces both vehicles to stop.

Since our algorithm is designed for situations, in which all other vehicles are moving in the same principal direction, we regard scenarios with no oncoming vehicles and no vehicles crossing the lane.

For the following simulations, parameters for the IDM are chosen as given in table I. The temporal interval for trajectory planning $[0, T]$, has a horizon of $T = 10\text{s}$ and is discretized by $N = 101$ nodes, leading to a temporal spacing of $\Delta t = T/(N - 1) = 100\text{ms}$. The method was implemented within C++ and running on a PC with an Intel-i7 CPU and 32GB of RAM. The OCP was solved by using the WORHP library for sequential quadratic programming [15]. It is to

mention, that due to the closed-form solvability of the OCP, one could also use less powerful solvers. The first scenario shows how the ego vehicle is driving on a winding rural road. The according results are depicted in figure 2. The trajectory is depicted as a sequence of points, where the spatio-temporal relation is color coded from blue to red.

It becomes obvious, that the trajectory starts from the ego vehicles position and smoothly approaches the centerline. Without acceleration and jerk terms in the OCP, just the spatial terms would be optimized and thus the trajectory would lie exactly on the centerline, except for the first three points, since they are fixed as explained above. We want to emphasize, that trajectories calculated directly by the IDM are in general not smooth and will consequently not lead to good tracking performances of vehicle controllers. Due to the smoothing effect of acceleration as well as jerk terms in the optimization problem, our trajectory gets smoothed and will lead to comfortable driving behavior. Figure 3 shows the velocity which increases in the beginning to approach the target speed v_0 and decreases subsequently due to the winding segment. The absolute acceleration which is given in figure 4 takes on an initial value of $\approx 2.5\text{m s}^{-2}$ and decreases then to $\approx 0.5\text{m s}^{-2}$, this is due to the initial longitudinal acceleration to quickly come up to the target speed and the lateral acceleration to steer back to the centerline. Braking acceleration is required to reduce the speed when getting closer to the winding road segment. The acceleration increases due to cornering and becomes smaller as soon as the trajectory again reaches the straight road segment.

In the next scenario, the focus is more on behavioral aspects: The ego vehicle drives behind a leading one. The traffic light, whose position is marked by the black stop line remains in a red state. Thus, both vehicles have to come to a stop before reaching the stop line. Replanning is performed every $t_{\text{repl.}} = 100\text{ms}$, hence the ego vehicle follows a trajectory for $t_{\text{repl.}}$ until a new one is calculated. To plan appropriate trajectories, knowledge of the IDM is used as follows: At first, the leading vehicles trajectory is predicted using the IDM. Therefore, the stop line is treated as a vehicle with zero velocity. In this case the interaction term of the IDM predominates the behavior and thus, the velocity decreases to zero. This predicted trajectory of the leading vehicle is then used to calculate the trajectory for the ego vehicle by also using the IDM.

Figure 5 shows snapshots of the scene for different time periods. As can be seen, while both vehicles are in motion, the ego vehicle keeps a safety distance to the leading vehicle. This distance decreases monotonously and finally converges to the standstill distance.

Figure 6 and 7 show the velocities and accelerations of the ego vehicle during the whole scenario duration of $t = 20\text{s}$. As depicted in figure 6, the ego vehicle starts with an higher velocity than the leading vehicle. Nevertheless, the IDM forces the ego vehicle to decelerate to even smaller velocities to maintain the sufficient safety distance. Therefore, as displayed in both figures, the ego vehicle brakes down rapidly

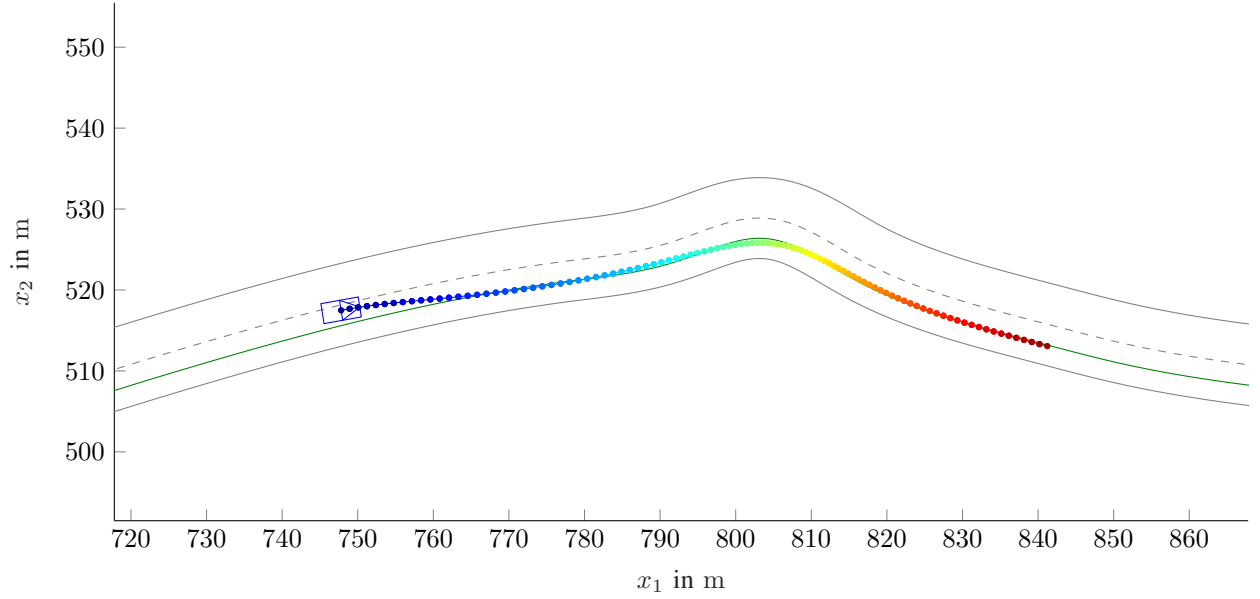


Fig. 2. Winding rural road scenario. The planned trajectory is color coded over time (from blue to red). The centerline used for planning is depicted in green.

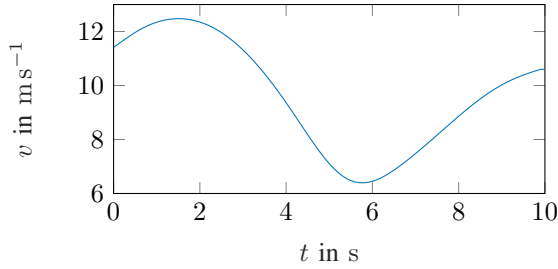


Fig. 3. Trajectory velocity.

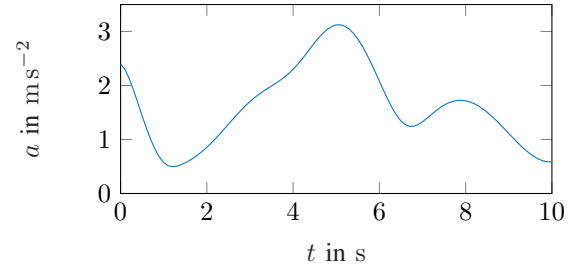


Fig. 4. Trajectory absolute acceleration.

TABLE II
MEASURED TIME NEEDED FOR CALCULATION

N	101	501	1001
$t_{\text{meas}}[\text{ms}]$	0.5792	6.7333	22.8417

in the very beginning of the scene.

This scenario also shows the advantage of incorporating model knowledge into the optimization process. While the trajectory planner presented in [3] is not designed for following or stopping, the calculation of a reference trajectory with a driver model allows the enforcement of different strategies using an almost identical optimization problem. In addition, the low computational costs enable interaction aware and foresighted behavior even for long horizons which is indispensable e.g. when driving towards a red traffic light with high velocity. As a result, the simulation shows that the presented concept is able to generate complex behavior with low computational effort.

The scenario was repeated for various numbers N of trajectory support points. The averaged computation times needed for planning are shown in table II. Evidently, the computation

time increases for higher numbers of trajectory support points. However, even for $N = 1001$, the run time is far below the cycle time in [3] and thus, the method is absolutely real-time capable. Such low run times can be realized, due to the comparably low computational burden for calculating IDM predictions and the subsequent optimization problem being quadratic allowing convergence in a single step using WORHP.

IV. CONCLUSION

In this work, we presented a trajectory planning approach for automated vehicles based on driver models. We used the intelligent driver model (IDM) to predict the behavior of all traffic participants, including the ego vehicle. Subsequently, a quadratic optimization problem containing appropriate acceleration and jerk terms is stated, to obtain smooth trajectories. The method is real-time capable, since the IDM prediction is fast to compute and the optimal control problem converges in one step using sequential quadratic programming. The benefit of using behavioral driver models is shown by means of a traffic light scenario, where both, the ego vehicle as well as the leading vehicle need to stop due to a red

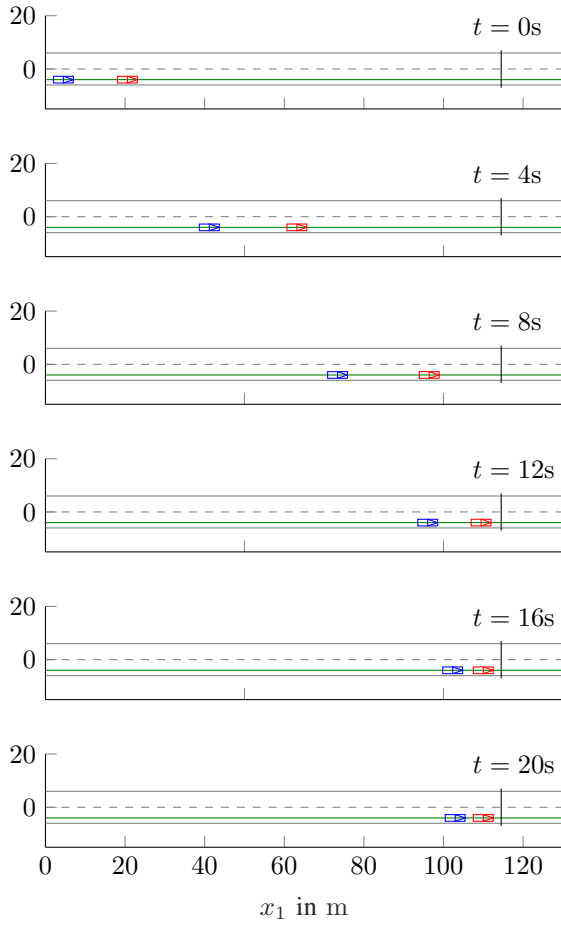


Fig. 5. Traffic light scenario: Both, the ego vehicle (blue) as well as the leading vehicle (red) need to stop in front of the red traffic light (indicated by the black stop line).

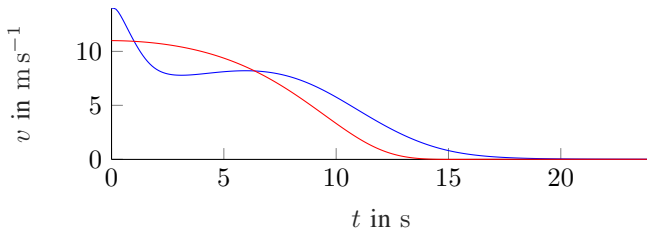


Fig. 6. Velocities of the ego vehicle (blue) and the leading vehicle (red) during the traffic light scenario.

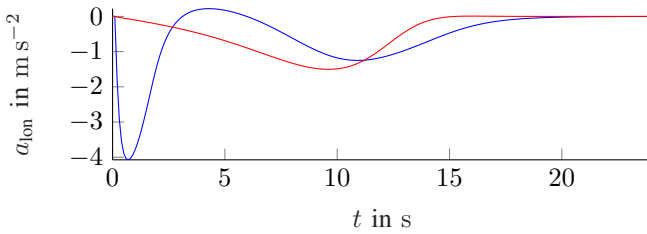


Fig. 7. Accelerations of the ego vehicle (blue) and the leading vehicle (red) during the traffic light scenario.

traffic light. We want emphasize, that the IDM may be easily exchanged by another model, like e.g. the foresighted driver model [13], that is able to handle even more complex scenarios. For this reason, future work includes the analysis and development of more advanced driver models. Thereby, for example uncertainties of perception modules can be taken into account as shown in [16] or [13].

ACKNOWLEDGMENT

This paper has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No 69075.

REFERENCES

- [1] C. Katrakazas *et al.*, "Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions," *Transportation Research Part C: Emerging Technologies*, vol. 60, pp. 416 – 442, 2015. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0968090X15003447>
- [2] B. Paden *et al.*, "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] J. Ziegler *et al.*, "Trajectory planning for bertha-a local, continuous method," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*. IEEE, 2014, pp. 450–457.
- [4] J. Ziegler *et al.*, "Making bertha drive-an autonomous journey on a historic route," *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 2, pp. 8–20, 2014.
- [5] M. Werling *et al.*, "Optimal trajectory generation for dynamic street scenarios in a frenet frame," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 987–993.
- [6] 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.
- [7] W. Lim *et al.*, "Behavioral trajectory planning for motion planning in urban environments," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Oct 2017, pp. 515–521.
- [8] 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.
- [9] P. Bender *et al.*, "The combinatorial aspect of motion planning: Maneuver variants in structured environments," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, June 2015, pp. 1386–1392.
- [10] F. Damerow and J. Eggert, "Risk-averse behavior planning under multiple situations with uncertainty," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, Sept 2015, pp. 656–663.
- [11] Ö. Ş. Taş *et al.*, "Making bertha cooperate-team annieway's entry to the 2016 grand cooperative driving challenge," *IEEE Transactions on Intelligent Transportation Systems*, 2017.
- [12] 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.
- [13] J. Eggert, F. Damerow, and S. Klingelschmitt, "The foresighted driver model," in *Intelligent Vehicles Symposium (IV), 2015 IEEE*. IEEE, 2015, pp. 322–329.
- [14] M. Liebner *et al.*, "Velocity-based driver intent inference at urban intersections in the presence of preceding vehicles," *IEEE Intelligent Transportation Systems Magazine*, vol. 5, no. 2, pp. 10–21, 2013.
- [15] C. Büskens and D. Wassel, "The esa nlp solver worhp," in *Modeling and optimization in space engineering*. Springer, 2012, pp. 85–110.
- [16] E. Ward *et al.*, "Probabilistic model for interaction aware planning in merge scenarios," *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 2, pp. 133–146, June 2017.