

Efficient Mixed-Integer Programming for Longitudinal and Lateral Motion Planning of Autonomous Vehicles

Christina Miller*, Christian Pek*, and Matthias Althoff

Abstract—The application of continuous optimization to motion planning of autonomous vehicles has enjoyed increasing popularity in recent years. In order to maintain low computation times, it is advantageous to have a convex formulation, in general requiring the planning problem to be separated into a longitudinal and lateral component. However, this decoupling of the motion often results in infeasible trajectories in situations in which both components need to be heavily linked, e.g., when planning swerving maneuvers to avoid a collision with obstacles. In this work, we propose an approach which extends the convex optimization problem of the longitudinal component to incorporate changing constraints, allowing us to guarantee feasibility of the resulting combined trajectory. Furthermore, we provide additional safety guarantees for the planned motion by integrating formal safety distances assuming infinite precision arithmetic. Our approach is demonstrated using simulated lane change maneuvers.

I. INTRODUCTION

Motion planning for autonomous vehicles aims to provide comfortable, feasible, and safe trajectories [1]. Different techniques have been proposed over the years to achieve this goal. We first review graph-based techniques, followed by continuous optimization.

Sampling-based trajectory planners, such as *Rapidly Exploring Random Trees* (RRT) [2], [3], randomly sample and connect states towards a goal region to obtain a kinematically feasible and collision-free trajectory. Its extension RRT* [4], [5] provides asymptotically optimal trajectories. However, both algorithms might not obtain jerk-optimal motions [6]. In [7], [8], so-called *state lattices* are planned. These are sets of trajectories whose goal states are vertices of a fixed predefined grid, resulting in a lattice structure. Sampling has been combined with optimal control in [9] to determine jerk-optimal trajectories by making use of quintic polynomials. The *State lattices* method provides good results, but lack optimality due to the fixed grid.

In order to overcome discretization effects, continuous optimization is often applied to robot motion planning [10]–[12]. The task of determining a feasible and collision-free trajectory is solved by minimizing a cost function with respect to a set of state and input constraints (and possibly a set of disturbances). For autonomous vehicles, the motion planning problem is formulated as a mixed-integer program

in [13] and as a non-linear optimization problem solved by sequential quadratic programming (SQP) in [14]. The resulting optimization problems are non-convex and thus harder to solve [15].

The computational complexity of the motion planning problem can be reduced by linearizing the non-linear, non-holonomic vehicle dynamics and separating the motion into a longitudinal and a lateral component [16]. As a result, the motion planning problem can be approximated with convex optimization programs, for which efficient solvers exist [17]. Convex collision avoidance approaches for autonomous vehicles are proposed in [18], [19]. Optimal longitudinal and lateral trajectories are obtained in [20], [21] using linear-quadratic programs (QP). In [22], a convex formulation is exploited to predict trajectories of traffic participants in multi-vehicle planning. Unfortunately, the longitudinal and lateral separation often results in infeasible trajectories (cf. Sec. II).

Recent approaches try to eliminate the problem of obtaining infeasible solutions using pre-planning and dexterous constraint formulations. For instance, a rough longitudinal motion is pre-planned in [23], which is used to determine a short-term lateral motion afterwards. Pre-planning a rough motion works well in simple scenarios, but comes to its limits when the feasibility of the lateral motion is highly linked to the planned longitudinal motion, e.g., when swerving is required to avoid a collision with obstacles. In order to consider the position constraints in such scenarios, convex safety regions are proposed in [24]. Each of the regions corresponds to different valid constraints imposed by safety-relevant obstacles. However, an efficient approach to determine these regions has not yet been provided.

II. MOTIVATION AND CONTRIBUTION

We initially consider a convex optimization problem, separated in longitudinal and lateral direction, to reduce the computational complexity. In order to obtain a feasible trajectory, the longitudinal and lateral planners have to respect the dynamics of the ego vehicle at any point in time (cf. friction circle [25, p. 382]). However, to satisfy dynamics, the longitudinal and lateral planners require prior knowledge of the intended future motion of the respective other planner. Fig. 1 illustrates the dependency of the two planners using an urban example: the ego vehicle has to steer through parked cars in a one-way road. In order to accomplish the task, the lateral planner has to determine a suitable lateral acceleration and apply it at a certain point in time t_1 to steer to the left side (in driving direction). On the

*The first two authors contributed equally to this work.

Christina Miller and Matthias Althoff are with the Technische Universität München, Fakultät für Informatik, Lehrstuhl für Robotik und Echtzeitsysteme, Boltzmannstraße 3, 85748, Garching, Germany. {christina.miller, althoff}@tum.de

Christian Pek is with BMW Group, D-85716 Unterschleißheim, Germany. Christian.Pek@bmw.de

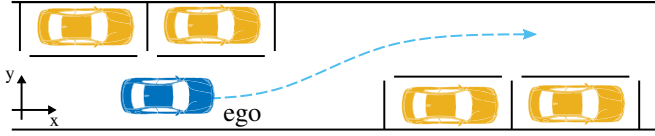


Fig. 1. Illustration of the feasibility problem. The ego vehicle (blue) has to pass parked vehicles.

other hand, the longitudinal planner must determine a point in time t_2 for starting to decelerate in order to not cause a collision with the parked car in front of the ego vehicle. The times t_1 and t_2 cannot be obtained independently. Additionally, the motion planners face changing position constraints. The longitudinal position is constrained by the parked vehicles at the beginning of the illustrated maneuver. After gaining enough lateral distance to pass the parked cars, this constraint is not relevant anymore. Integrating these changing constraints into existing optimization tools in a simple yet efficient manner is challenging.

Contribution: In this work, we propose an efficient mixed-integer programming approach to resolve the issues of re-combining the longitudinal and lateral motion of convex motion planners, which

- 1) ensures feasibility of the lateral motion profile in situations with changing position constraints,
- 2) can be easily integrated by extending the optimization vector of the convex longitudinal optimization problem, and
- 3) provides safety guarantees by incorporating formal safety distances.

Our approach works as follows (cf. Fig. 2): we divide the planned maneuver in safe convex regions (cf. [26]) in the time domain based on the intended lateral motion. For each region, we specify longitudinal position constraints (including formal dynamic safety distances) for the ego vehicle and a binary encoding of this region in order to integrate the changing constraints into the longitudinal

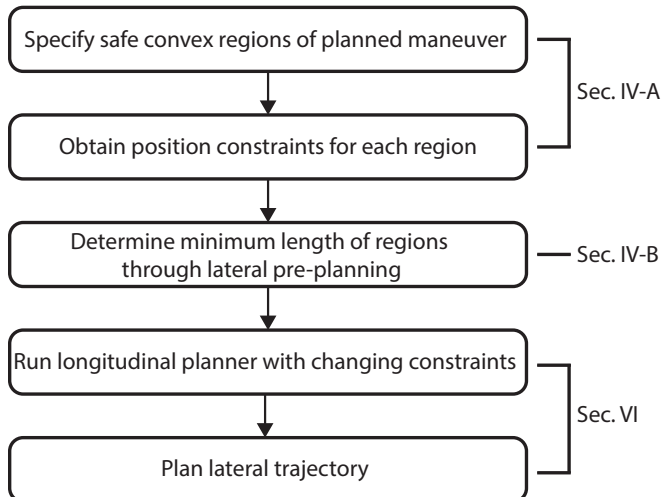


Fig. 2. Procedure of the proposed approach to compute feasible trajectories.

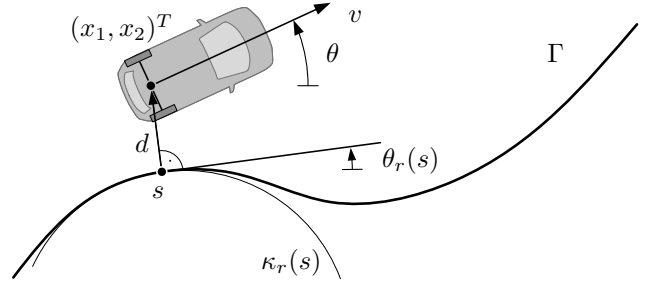


Fig. 3. Kinematics of the vehicle's motion in path coordinates [21].

planner. Using pre-planning of a lateral trajectory, we can determine the minimum length of the regions to guarantee feasibility. During the longitudinal planning, these safe convex regions (required by the lateral planner) are optimally distributed across the optimization horizon to perform the desired maneuver. After obtaining the longitudinal motion, the lateral motion planner is able to determine a feasible lateral motion.

This paper is structured as follows: Sec. III introduces the utilized models and assumptions. Afterwards, the extension of the underlying optimization problem to incorporate changing constraints is described in Sec. IV. The integration of formal safety distances is explained in Sec. V. This paper finishes with a validation of the presented approach using simulated lane change maneuvers in Sec. VI followed by conclusions in Sec. VII.

III. PRELIMINARIES

Without loss of generality, we describe the vehicle's kinematics in path coordinates [9], i.e., Cartesian poses $(x_1, x_2, \theta)^T \in \mathbb{R}^3$ are replaced by $(s, d, \theta)^T \in \mathbb{R}^3$ in a curvilinear coordinate system aligned with the center Γ of the lane the ego vehicle is initially driving in, defined by the curvature κ_r and the reference orientation θ_r (cf. Fig. 3). A lane change can be modeled using an offset to the reference Γ . We use $v \geq 0$ to describe the velocity and a to describe the acceleration of a vehicle.

We assume that the initial time is $t_0 = 0$. In addition, the following time-invariant assumptions are made:

- 1) the positions and velocities of safety-relevant vehicles are fully observable,
- 2) the maximum absolute acceleration $|a_{\max}|$ of vehicles is known, and
- 3) a prediction of the future motion of vehicles is given.

Note that our approach can also be extended to include fail-safe trajectories to guarantee safety even if vehicles vigorously deviate from the predicted motion [27].

To focus on the novel aspects, we only consider yielding maneuvers, in particular lane changes. However, our approach is applicable to other types of maneuvers as well.

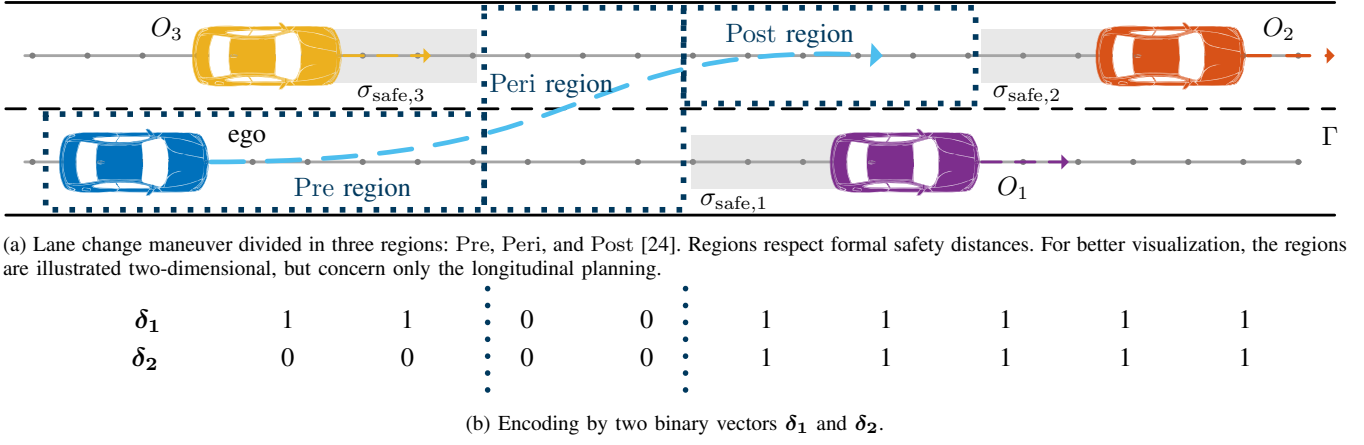


Fig. 4. Lane change maneuver divided in regions, encoded by binary vectors.

IV. COMBINED LONGITUDINAL AND LATERAL TRAJECTORIES

We model different stages of the lateral motion during a maneuver, e. g., lane following and subsequent swerving, by constructing safe convex regions within the free space of the ego vehicle. Integrating these regions as changing constraints into the optimization problem of the longitudinal planner enables us to guarantee feasibility of the planned trajectory.

A. Formulation of Safe Convex Regions

During a maneuver, the longitudinal position of the ego vehicle is constrained by different surrounding vehicles and objects depending on the lateral motion of the ego vehicle.

We introduce $N_{\text{long}} \in \mathbb{N}$ as the finite discrete optimization horizon of the longitudinal planner. Inspired by [24], we denote the different stages (depicted in Fig. 4a) of a lane change maneuver as Pre (following the initial lane), Peri (lane change), and Post (following the goal lane). We formally define them for the optimization step $k \in \{0, \dots, N_{\text{long}}\}$ as

$$\begin{aligned}
 \text{Pre}^k &= \{s_{\text{ego}} \in \mathbb{R} \mid s_{\text{ego}} < s_1(k) - \sigma_{\text{safe},1}\}, \\
 \text{Peri}^k &= \{s_{\text{ego}} \in \mathbb{R} \mid s_{\text{ego}} < s_1(k) - \sigma_{\text{safe},1} \wedge \\
 &\quad s_{\text{ego}} < s_2(k) - \sigma_{\text{safe},2} \wedge \\
 &\quad s_{\text{ego}} > s_3(k) + \sigma_{\text{safe},3}\}, \\
 \text{Post}^k &= \{s_{\text{ego}} \in \mathbb{R} \mid s_{\text{ego}} < s_2(k) - \sigma_{\text{safe},2}\},
 \end{aligned} \tag{1}$$

where $s_i \in \mathbb{R}, i \in \{1, 2, 3\}$ correspond to the predicted longitudinal positions and $\sigma_{\text{safe},i}$ to the safety distance to the surrounding vehicles (cf. Fig. 4a). Note that other maneuvers may require the construction of fewer or more convex regions or a different number of vehicles.

If the Peri region is small with considering the minimum and maximum allowed position of the ego vehicle, the resulting motion of the lane change requires high lateral accelerations. To limit the required maximum lateral acceleration, we enforce the Peri region to have a pre-computed minimum length $n_{\text{min}} \in \{0, \dots, N_{\text{long}}\}$ at each optimization step k . Note that n_{min} becomes smaller at each

re-planning cycle (cf. Sec. IV-B) and eventually becomes zero when the ego vehicle has fully entered the goal lane.

To formulate this as a mixed-integer program, we define a binary encoding (cf. Fig. 4b) of the three regions using two binary vectors, $\delta_1 \in \{0, 1\}^{N_{\text{long}}}$ and $\delta_2 \in \{0, 1\}^{N_{\text{long}}}$.

Definition 1: The binary encoding of the three regions in (1) is

$$[\delta_1(k) \quad \delta_2(k)] = \begin{cases} [1 \ 0] & s_{\text{ego}}(k) \in \text{Pre}^k \\ [0 \ 0] & s_{\text{ego}}(k) \in \text{Peri}^k \\ [1 \ 1] & s_{\text{ego}}(k) \in \text{Post}^k. \end{cases}$$

Since the temporal order¹ of the regions is given for the maneuver, the values of δ_1 and δ_2 can be further constrained.

Definition 2: The binary encoding in Def. 1 must satisfy the following constraints:

- 1) δ_2 switches from zero to one at most once;
- 2) δ_2 does not switch before δ_1 ;
- 3) After the Peri region, δ_1 and δ_2 switch jointly;
- 4) The Peri region is at least $n_{\text{min}} \in \{0, \dots, N_{\text{long}}\}$ long, which can be written as $N - \sum_{k=1}^N \delta_1(k) \geq n_{\text{min}}$.

Depending on the region the ego vehicle is located in, different position constraints must be active (cf. (1)). We use the Big-M method [28, pp. 156–159] to include these changing constraints with respect to the binary encoding (cf. Def. 1) into the optimization problem of the longitudinal planner. After introducing the resulting linear constraints in (2), we briefly recall the Big-M method.

$$\begin{aligned}
 s_{\text{ego}}(k) &\leq s_{O_1}(k) - \sigma_{\text{safe},1}(k) + \delta_2(k)M_{\text{big}} \\
 s_{\text{ego}}(k) &\leq s_{O_2}(k) - \sigma_{\text{safe},2}(k) + (\delta_1(k) - \delta_2(k))M_{\text{big}} \\
 s_{\text{ego}}(k) &\geq s_{O_3}(k) + \sigma_{\text{safe},3}(k) - \delta_1(k)M_{\text{big}}.
 \end{aligned} \tag{2}$$

By use of a sufficiently large constant M_{big} , constraints can be activated and deactivated at certain discrete points in time within the optimization horizon [29]. The value of M_{big}

¹Modeling situations in which a maneuver needs to be aborted is done by reversing the temporal order for the next planning cycle.

must be large enough to not restrict feasible solutions of the optimization problem, but small enough to not introduce numerical instabilities during solving. If the factors of M_{big} in (2) evaluate to zero, the variable M_{big} is eliminated and the constraints are active. If the binary variables evaluate to one, the equations are always fulfilled due to the large value of M_{big} and therefore, the respective constraints are deactivated; except for the Peri region constraint due to the utilized encoding. The advantage of using the Big-M method is that linear constraints remain linear. Thus, we can include the changing constraints directly into the linear-quadratic optimization problem.

The presented approach is independent of the formulation of the lateral and longitudinal optimization problems and their cost functions, since the optimization vector is merely extended, but not changed. As an example of the longitudinal optimization problem, we extend a linear-quadratic optimization problem

$$\begin{aligned} \arg \min_u \quad & J(u) \\ \text{s.t.} \quad & x(\hat{k} + 1) = Ax(\hat{k}) + Bu(\hat{k}), \\ & x(0) = x_0, \\ & u_{\min}(k) \leq u(k) \leq u_{\max}(k), \\ & x_{\min}(k) \leq x(k) \leq x_{\max}(k), \end{aligned}$$

with an initial state x_0 , a quadratic cost function $J(u)$, a linear vehicle model with state matrix A and input matrix B as well as state and input constraints for discrete points in time $k \in \{1, \dots, N\}$ and $\hat{k} \in \{1, \dots, N-1\}$.

Similar to [29], we extend the optimization vector u to $\tilde{u} = [u, \delta_1, \delta_2]$ and adapt the cost function to obtain the mixed-integer quadratic problem

$$\begin{aligned} \arg \min_{\tilde{u}} \quad & \tilde{J}(\tilde{u}) \\ \text{s.t.} \quad & x(\hat{k} + 1) = Ax(\hat{k}) + Bu(\hat{k}), \\ & x(0) = x_0, \\ & u_{\min}(k) \leq u(k) \leq u_{\max}(k), \\ & x_{\min}(k) \leq x(k) \leq x_{\max}(k), \\ & s_{\text{ego}}(k) \leq s_{O_1}(k) - \sigma_{\text{safe},1}(k) + \delta_2(k)M_{\text{big}}, \\ & s_{\text{ego}}(k) \leq s_{O_2}(k) - \sigma_{\text{safe},2}(k) \\ & \quad + (\delta_1(k) - \delta_2(k))M_{\text{big}}, \\ & s_{\text{ego}}(k) \geq s_{O_3}(k) + \sigma_{\text{safe},3}(k) - \delta_1(k)M_{\text{big}}, \\ & \delta_2(\hat{k}) \leq \delta_2(\hat{k} + 1), \\ & \delta_2(k) \leq \delta_1(k), \\ & \delta_2(\hat{k}) - \delta_1(\hat{k}) \leq \delta_2(\hat{k} + 1) - \delta_1(\hat{k} + 1), \\ & N - \sum_{k=1}^N \delta_1(k) \geq n_{\min} \in \{0, \dots, N\}, \end{aligned}$$

where the first block of constraints corresponds to the original optimization problem, the second block describes

the previous defined regions (2), and the third block ensures the correctness of the binary vectors according to Def. 2.

B. Feasibility Guarantee of Trajectories

In this subsection, we prove that we can choose the minimum length n_{\min} of the Peri region (cf. constraint 4 in Def. 2) during the longitudinal planning so that the existence of a feasible lateral profile is guaranteed.

In the beginning of each re-planning cycle, a lateral trajectory is planned with respect to a parameterized longitudinal profile near the maximum acceleration $|a_{x,\max}| < |a_{\max}|$. Based on the obtained lateral trajectory, one can extract the minimum length of the Peri region. This trajectory is solely used for determining n_{\min} , which is then used for the subsequent planning of the longitudinal motion to guarantee the existence of a lateral trajectory.

Let us first consider the lateral dynamics of the ego vehicle during the lane change. Given some longitudinal acceleration a_x , we can determine the maximum feasible lateral acceleration a_y with respect to the friction circle.

Definition 3: The maximum feasible lateral acceleration $a_{y,\text{feasible}}(a_x)$ given some longitudinal acceleration $a_x < a_{\max}$ according to the friction circle [25, p. 382] corresponds to

$$a_{y,\text{feasible}}(a_x) := \sqrt{a_{\max}^2 - a_x^2}.$$

Based on the feasible lateral acceleration, we can compute the time required to perform a swerving maneuver to an adjacent lane, including some reaction time δ_{steer} for the steering actuators.

Definition 4: The minimum time t_{\min} required to traverse the lateral distance d , given a lateral acceleration a_y and reaction time δ_{steer} , corresponds to (cf. [30]):

$$t_{\min}(a_y) := \sqrt{\frac{2d}{a_y}} + \delta_{\text{steer}}.$$

Assuming that the longitudinal planning algorithm does not plan longitudinal trajectories with acceleration values near the maximum acceleration a_{\max} (justified if the longitudinal planner uses safety margins in the acceleration), the following theorem holds:

Theorem 1: Given $a_{x,\max}^2 + a_{y,\max}^2 \leq a_{\max}^2$, there exists a number $n_{\min} \in \{0, \dots, N_{\text{long}}\}$ of time steps so that for each longitudinal trajectory satisfying constraint 4 in Def. 2, a feasible lateral trajectory exists.

Proof: $a_{y,\text{feasible}}(a_x)$ (cf. Def. 3) decreases monotonically with increasing longitudinal acceleration a_x . Therefore, since $a_{x,\max} < a_{\max}$, the lowest maximum feasible lateral acceleration $\underline{a}_{y,\text{feasible}} := a_{y,\text{feasible}}(a_{x,\max}) > 0$ is always feasible for all trajectories that satisfy $a_{x,\max}$.

Since t_{\min} (cf. Def. 4) is monotonically decreasing with increasing $a_{y,\text{feasible}}$, the maximum required traverse time is $\bar{t}_{\min} := t_{\min}(\underline{a}_{y,\text{feasible}})$. We define $n_{\min} = \lceil \bar{t}_{\min} / \Delta t_{\text{long}} \rceil$ for the longitudinal planning step size Δt_{long} . Let $a_x(k)$, $k \in \{1, \dots, N\}$ be a longitudinal acceleration

profile of a trajectory with $a_x(k) \leq a_{x,\max} \quad \forall k$. Then, $a_{y,\text{feasible}}(a_x(k)) \geq \underline{a}_{y,\text{feasible}}$ and $t_{\min}(a_{y,\text{feasible}}(a_x(k))) \leq \bar{t}_{\min} \leq n_{\min} \Delta t_{\text{long}}$ hold for all k . ■

If $n_{\min} > N$, we set $n_{\min} = N$. While performing the intended maneuver, the remaining required lateral distance d decreases. Therefore, t_{\min} and consequently n_{\min} decrease over time when re-planning (cf. Def. 4) and eventually become zero.

V. INTEGRATION OF FORMAL SAFETY DISTANCES

According to the Vienna Convention on Road Traffic, it is the duty of the vehicles to keep a “sufficient distance [...] to avoid [a] collision if the vehicle in front should suddenly slow down or stop” [31, §13]. These distances are formalized using the maximum acceleration of the ego vehicle $a_{\max,\text{ego}}$ and the one of the preceding vehicle $a_{\max,\text{obj}}$ [30], [32] as

$$\begin{aligned} \sigma_{\text{safe}}^a &= \frac{(v_{\text{obj}} - |a_{\max,\text{obj}}| \delta_{\text{brake}} - v_{\text{ego}})^2}{-2(|a_{\max,\text{obj}}| - |a_{\max,\text{ego}}|)} - v_{\text{obj}} \delta_{\text{brake}} \\ &\quad + \frac{1}{2} |a_{\max,\text{obj}}| \delta_{\text{brake}}^2 + v_{\text{ego}} \delta_{\text{brake}}, \\ \sigma_{\text{safe}}^b &= \frac{v_{\text{obj}}^2}{-2|a_{\max,\text{obj}}|} - \frac{v_{\text{ego}}^2}{-2|a_{\max,\text{ego}}|} + v_{\text{ego}} \delta_{\text{brake}}, \end{aligned} \quad (3)$$

where δ_{brake} is the maximum reaction time of the braking actuators and v_{ego} and v_{obj} are the velocities of the ego vehicle and preceding vehicle, respectively. Note that depending on the vehicles’ velocities and maximum decelerations either σ_{safe}^a or σ_{safe}^b has to be used according to [32]. In case we want to compute the safety distance to a following vehicle, the ego vehicle is assumed to be the preceding vehicle.

The velocities of the preceding vehicles are provided by the used prediction. The safety distance is quadratic in the velocity of the ego vehicle, thus, safety distances cannot be directly included in optimization programs with linear constraints. To circumvent this problem, we exploit the convexity of the safety distance and use a piecewise linear approximation for both distances in (3) instead. The resulting linear approximation of the safety distance is over-approximative and therefore does not impair safety.

We use p linear functions $g_1, g_2, \dots, g_p : \mathbb{R} \rightarrow \mathbb{R}$ to approximate the safety distance $\sigma_{\text{safe}} \in \{\sigma_{\text{safe}}^a, \sigma_{\text{safe}}^b\}$. To achieve this, we divide the valid velocity range $[v_{\min}, v_{\max}]$

in p equally large intervals $[v_i, v_{i+1}]$, $i \in \{0, \dots, p-1\}$ by setting

$$v_i = (v_{\max} - v_{\min}) \frac{i}{p} + v_{\min}. \quad (4)$$

For each interval, the safety distance is approximated using a linear function, resulting in the safety distance formulation

$$\tilde{\sigma}_{\text{safe}}(v_{\text{ego}}) = \begin{cases} g_1(v_{\text{ego}}), & v_0 \leq v_{\text{ego}} < v_1, \\ g_2(v_{\text{ego}}), & v_1 \leq v_{\text{ego}} < v_2, \\ \vdots \\ g_p(v_{\text{ego}}), & v_{\text{ego}} \geq v_{p-1}. \end{cases} \quad (5)$$

In Fig. 5, a piecewise linear approximation of the safety distance is shown. The ego vehicle is not allowed to enter the shaded region in order to guarantee safety in case the leading vehicle performs an emergency braking.

In order to integrate the p linear functions into the optimization problem, we make use of the fact that each convex, piecewise linear function can be represented as a maximum function [33]. Thus, the convex safety distance can be reformulated as

$$\tilde{\sigma}_{\text{safe}}(v_{\text{ego}}) = \max \{g_1(v_{\text{ego}}), g_2(v_{\text{ego}}), \dots, g_p(v_{\text{ego}})\}.$$

Respecting the maximum of these p linear functions is equivalent to satisfying every single one of them due to convexity.

According to the unique general form of linear equations, each linear function $g_i(v_{\text{ego}})$ can be expressed as $g_i(v_{\text{ego}}) = m_i v_{\text{ego}} + \Delta_i$ allowing us to further rearrange the constraint (2) for vehicle O_1 to

$$s_{\text{ego}} \leq s_{O_1} - (m_i v_{\text{ego}} + \Delta_i) + \delta_2 M_{\text{big}}. \quad (6)$$

The reformulation for vehicles O_2 and O_3 works analogously. As a result, we obtain p linear position constraints for each surrounding vehicle. Larger numbers of linear functions p decrease the approximation error, but increase the computational time of solving the optimization problem.

VI. SIMULATION RESULTS

Our approach is implemented in MATLAB/Simulink and tested within our own simulation environment on an Intel i7-6820 processor using lane change maneuvers with speeds of up to 36.1 m/s. The commercial solver Gurobi [17] is used to solve the mixed-integer quadratic program.

The underlying longitudinal and lateral trajectory planning problems are modeled according to [21] and extended to our approach (cf. Sec. IV and Sec. V). In each planning cycle, a longitudinal trajectory is planned with an optimization horizon of $N_{\text{long}} = 20$ and a step size of $\Delta t_{\text{long}} = 0.5$ s. Afterwards, the lateral profile is planned with a horizon of $N_{\text{lat}} = 20$ and a step size of $\Delta t_{\text{lat}} = 0.25$ s with respect to the calculated longitudinal trajectory.

Fig. 6 shows an example lane change scenario, in which the ego vehicle is initially following the slower preceding vehicle O_1 and plans a trajectory into a small gap between

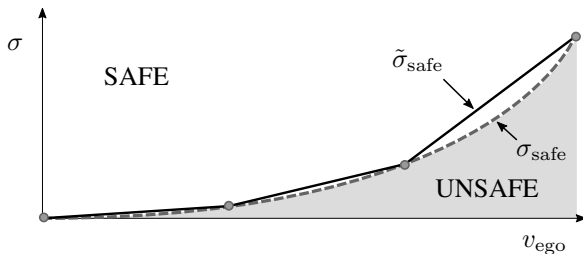


Fig. 5. Piecewise linear approximation $\tilde{\sigma}_{\text{safe}}$ (solid) of the convex safety distance σ_{safe} (dashed).

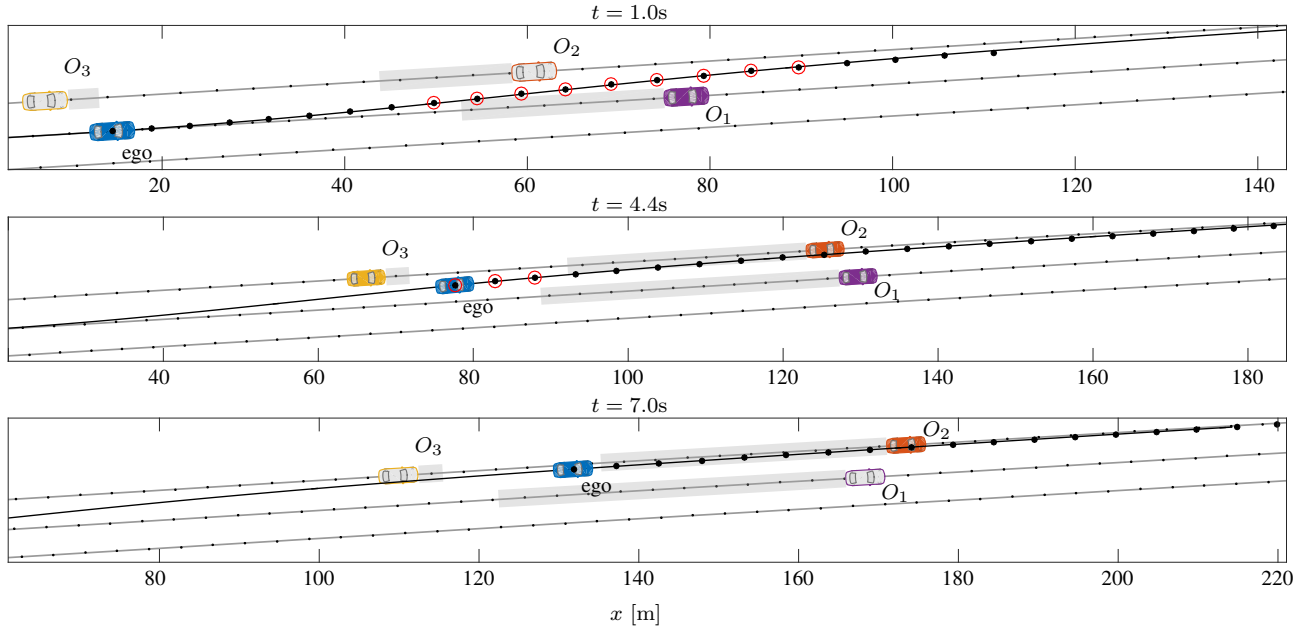


Fig. 6. Ego vehicle performs lane change maneuver into a small gap between two vehicles at three points in time, each in one of the regions Pre, Peri, and Post. The planned Peri region is illustrated with circles around the dots of the optimal trajectory. The driven trajectory (solid black) and reference curves (gray with black dots) are depicted. The vehicles that are set as active constraints are filled and deactivated ones are colored in gray.

the two vehicles O_2 and O_3 . The parameters are listed in Tab. I. We assume constant acceleration to predict the surrounding vehicles over the optimization horizon. Example points in time $t \in \{1.0\text{s}, 4.4\text{s}, 7.0\text{s}\}$ show the ego vehicle in the regions Pre, Peri, and Post. We use the step size $\Delta t_{\text{lat}} = 0.25\text{s}$ for the optimal trajectory in Fig. 6. Circled dots of the optimal trajectory denote the Peri region, obtained by our planner. Vehicles that are not set as active constraints in the optimization problem are colored in gray. At $t = 1.0\text{s}$, the constraints imposed by the vehicles O_2 and O_3 on the adjacent lane are deactivated.

The planned Peri region satisfies the pre-determined minimum length of $n_{\min} = 4$ according to the longitudinal discretization. At $t = 4.4\text{s}$, the ego vehicle is located in the Peri region and therefore has to consider all three surrounding vehicles. The remaining time for the lane change

is $n_{\min} = 1 \hat{=} 0.5\text{s}$, which is denoted by the three circled dots in the planning horizon at 0s , 0.25s , and 0.5s . Finally, the ego vehicle is shown in the Post region at $t = 7.0\text{s}$.

In Fig. 7 the optimal longitudinal position is illustrated at the same three points in time $t \in \{1.0\text{s}, 4.4\text{s}, 7.0\text{s}\}$. For improved visibility, only the first 10 optimization steps are plotted. Again, inactive constraints are colored in gray. At $t = 1.0\text{s}$, the planner optimizes the temporal start and end of the Peri region along the planning horizon such that $n_{\min} = 4$ and the considered safety distances are satisfied while respecting longitudinal dynamics. Before and after the Peri region, constraints of unconsidered vehicles are deactivated and thus the ego vehicle is allowed to enter the safety distances. At $t = 4.4\text{s}$, the ego vehicle has passed vehicle O_3 . The minimum required length of the Peri region has shrunk to $n_{\min} = 1$. Afterwards, the ego vehicle is able to pass O_1 and follow O_2 . At $t = 7.0\text{s}$, O_1 is also passed and $n_{\min} = 0$. Therefore, the position of the ego vehicle is only constrained by O_2 . Note that the planner tries to make the Peri region as long as possible at any time with the aim of increasing comfort. For this reason, the planned length of the Peri region is 2.5s at $t = 1.0\text{s}$ instead of the minimum required length of 2.0s .

All tested trajectories are feasible, which means that the lateral algorithm returns a drivable trajectory based on the given longitudinal velocity profile. The longitudinal planner is forced to plan a region with length n_{\min} in which all vehicles are considered. In Fig. 7, it is shown that this region is considered during the whole maneuver. In Fig. 6, one observes that the start and end of this region are planned consistently, which can be determined by the end of the

TABLE I
PARAMETERS OF THE PRESENTED LANE CHANGE SCENARIO

Variable	Value	Description
τ	40 ms	simulation cycle time
$v_{\text{ego, init}}$	16.67 m/s	initial velocity ego
v_1	15.28 m/s	velocity O_1
v_2	18.89 m/s	velocity O_2
v_3	17.22 m/s	velocity O_3
$a_{\text{max, ego}}$	4 m/s ²	absolute maximum acceleration ego
$a_{\text{max, obj}}$	8 m/s ²	absolute maximum acceleration surrounding vehicles
$d_{\text{gap, init}}$	29.9 m	initial gap size between O_2 and O_3 incl. safety distances
$n_{\min, \text{init}}$	4	initial required length of Peri region

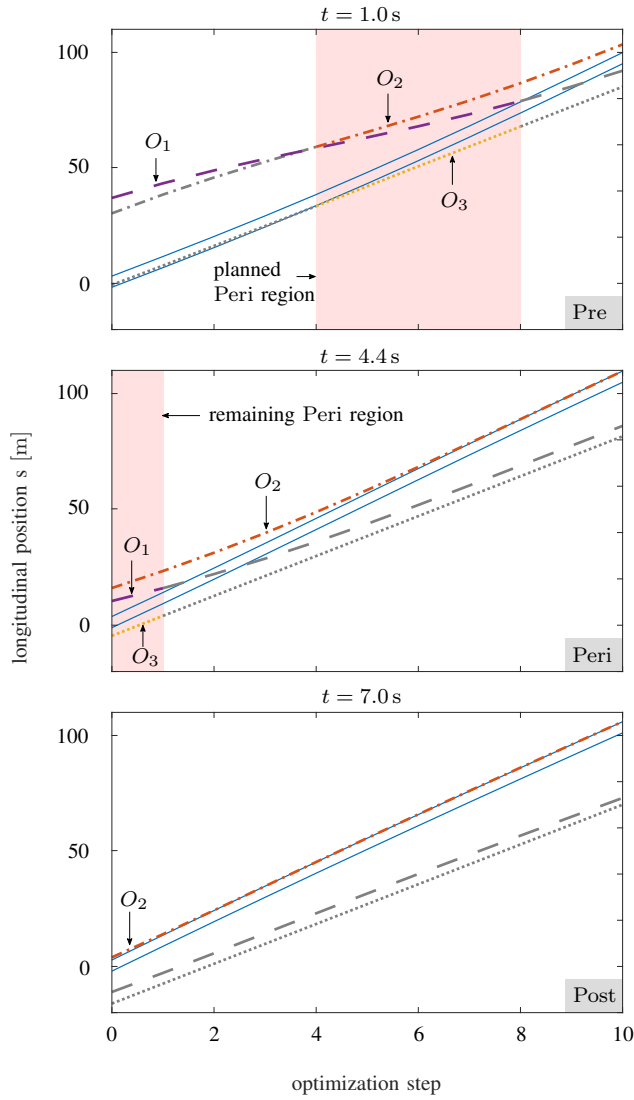


Fig. 7. Planned longitudinal position trajectory at three different points in time, each in one of the regions Pre, Peri, and Post. The ego vehicle's front and rear bumpers are represented in solid blue with the respective safety distances to surrounding vehicles (dashed). Inactive safety distances are colored in gray.

Peri region at around 85m in both time steps $t = 1.0$ s and $t = 4.4$ s.

We compare our approach against a sampling-based trajectory planner [9], in which trajectories are sampled for different end states using a quintic polynomial. Both planners require computation times of less than 0.1s. We use a cycle time of 0.1 s and sample in the time domain with a step size of $\Delta_t = 0.1$ s and in the position domain with $\Delta_{\text{long}} = 0.1$ m and $\Delta_{\text{lat}} = 0.5$ m in longitudinal and lateral direction. Both approaches obtain jerk-optimal trajectories. However, the sampling approach produces discontinuities when connecting subsequent planning cycles as shown in Fig. 8 for the scenario in Tab. I.

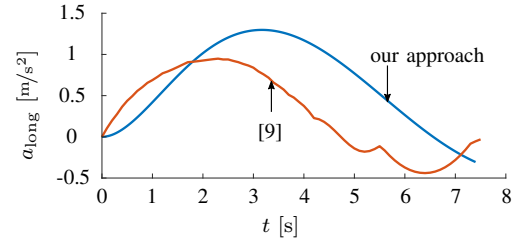


Fig. 8. Acceleration profile resulting from the presented approach (blue) and [9] (orange).

VII. CONCLUSIONS

Separating the motion planning task in lateral and longitudinal planning allows the formulation of two convex optimization problems but leads often to infeasible lateral trajectories. We guarantee the feasibility of lateral motions of autonomous vehicles during planning of a longitudinal motion by formulating an efficient mixed-integer program with a convex relaxation. By extending the optimization vector, the benefits of the original convex optimization problem can be preserved while the missing dependency between longitudinal and lateral directions is modeled. By pre-planning a lateral trajectory, we are able to determine the minimum time that is necessary in order to perform the lateral motion of the maneuver. Our approach explicitly considers the changing constraints during a maneuver as hard constraints during the optimization. The safety of the resulting trajectory is ensured by incorporating formal safety distances.

We have demonstrated our approach using simulated lane change maneuvers. Our approach is independent of the longitudinal and lateral planners and can be adapted to other types of maneuvers.

ACKNOWLEDGMENTS

This work is partially funded by the German Federal Ministry of Economics and Technology through the research initiative Ko-HAF (<https://www.ko-haf.de/>).

REFERENCES

- [1] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 33–55, 2016.
- [2] S. M. LaValle and J. J. Kuffner, "Randomized kinodynamic planning," *Intl. Journal of Robotics Research*, vol. 20, no. 5, pp. 378–400, 2001.
- [3] E. Frazzoli, M. A. Dahleh, and E. Feron, "Real-time motion planning for agile autonomous vehicles," in *Proc. of the American Control Conference*, 2001, pp. 43–49.
- [4] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Int. Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011.
- [5] Y. Kuwata, J. Teo, G. Fiore, S. Karaman, E. Frazzoli, and J. P. How, "Real-time motion planning with applications to autonomous urban driving," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 5, pp. 1105–1118, 2009.
- [6] D. González, J. Pérez, V. Milanés, and F. Nashashibi, "A review of motion planning techniques for automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1135–1145, 2016.

- [7] F. von Hundelshausen, M. Himmelsbach, F. Hecker, A. Mueller, and H.-J. Wuensche, "Driving with tentacles: Integral structures for sensing and motion," *Int. Journal of Field Robotics*, vol. 25, no. 9, pp. 640–673, 2008.
- [8] J. Ziegler and C. Stiller, "Spatiotemporal state lattices for fast trajectory planning in dynamic on-road driving scenarios," in *Proc. of the IEEE Int. Conf. on Intelligent Systems and Robots*, 2009, pp. 1879–1884.
- [9] M. Werling, S. Kammel, J. Ziegler, and L. Gröll, "Optimal trajectories for time-critical street scenarios using discretized terminal manifolds," *Int. Journal of Robotic Research*, vol. 31, no. 3, pp. 346–359, 2012.
- [10] N. Ratliff, M. Zucker, J. A. Bagnell, and S. Srinivasa, "CHOMP: Gradient optimization techniques for efficient motion planning," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2009, pp. 489–494.
- [11] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt, M. Sokolsky, G. Stanek, D. Stavens, A. Teichman, M. Werling, and S. Thrun, "Towards fully autonomous driving: Systems and algorithms," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2011, pp. 163–168.
- [12] D. Berenson, J. Kuffner, and H. Choset, "An optimization approach to planning for mobile manipulation," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2008, pp. 1187–1192.
- [13] T. Schouwenaars, B. De Moor, E. Feron, and J. How, "Mixed integer programming for multi-vehicle path planning," in *Proc. of the IEEE European Control Conference*, 2001, pp. 2603–2608.
- [14] J. Ziegler, P. Bender, T. Dang, and C. Stiller, "Trajectory planning for Bertha – a local, continuous method," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2014, pp. 450–457.
- [15] D. P. Bertsekas, *Nonlinear programming*. Athena Scientific, 2016.
- [16] P. Falcone, M. Tufo, F. Borrelli, J. Asgari, and H. E. Tseng, "A linear time varying model predictive control approach to the integrated vehicle dynamics control problem in autonomous systems," in *Proc. of the IEEE Int. Conf. on Decision and Control*, 2007, pp. 2980–2985.
- [17] I. Gurobi Optimization, "Gurobi optimizer reference manual," 2016. [Online]. Available: <http://www.gurobi.com>
- [18] S. J. Anderson, S. C. Peters, T. E. Pilutti, and K. Iagnemma, "An optimal-control-based framework for trajectory planning, threat assessment, and semi-autonomous control of passenger vehicles in hazard avoidance scenarios," *Int. Journal of Vehicle Autonomous Systems*, vol. 8, no. 2–4, pp. 190–216, 2010.
- [19] J. Nilsson, M. Ali, P. Falcone, and J. Sjöberg, "Predictive manoeuvre generation for automated driving," in *Proc. of the IEEE Int. Conf. on Intelligent Transportation Systems*, 2013, pp. 418–423.
- [20] B. Gutjahr, L. Gröll, and M. Werling, "Lateral vehicle trajectory optimization using constrained linear time-varying MPC," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 6, pp. 1586–1595, 2017.
- [21] B. Gutjahr, C. Pek, L. Gröll, and M. Werling, "Efficient trajectory optimization for vehicles using quadratic programming," *at - Automatisierungstechnik*, vol. 64, no. 10, p. 786–794, 2016.
- [22] J. Schulz, K. Hirsenkorn, J. Löchner, M. Werling, and D. Burschka, "Estimation of collective maneuvers through cooperative multi-agent planning," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2017, pp. 624–631.
- [23] W. Zhan, J. Chen, C.-Y. Chan, C. Liu, and M. Tomizuka, "Spatially-partitioned environmental representation and planning architecture for on-road autonomous driving," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2017, pp. 632–639.
- [24] J. Nilsson, M. Brännström, J. Fredriksson, and E. Coelingh, "Longitudinal and lateral control for automated yielding maneuvers," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 5, pp. 1404–1414, 2016.
- [25] R. Rajamani, *Vehicle dynamics and control*. Springer, 2012.
- [26] R. Deits and R. Tedrake, "Efficient mixed-integer planning for UAVs in cluttered environments," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2015, pp. 42–49.
- [27] S. Magdici and M. Althoff, "Fail-safe motion planning of autonomous vehicles," in *Proc. of the IEEE Int. Conf. on Intelligent Transportation Systems*, 2016, pp. 452–458.
- [28] I. Griva, S. G. Nash, and A. Sofer, *Linear and nonlinear optimization*. Siam, 2009.
- [29] X. Qian, F. Althé, P. Bender, C. Stiller, and A. de La Fortelle, "Optimal trajectory planning for autonomous driving integrating logical constraints: An MIQP perspective," in *Proc. of the IEEE Int. Conf. on Intelligent Transportation Systems*, 2016, pp. 205–210.
- [30] C. Pek, P. Zahn, and M. Althoff, "Verifying the safety of lane change maneuvers of self-driving vehicles based on formalized traffic rules," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2017, pp. 1477–1483.
- [31] Economic Commission for Europe: Inland Transport Committee. (1968) Vienna convention on road traffic. [Online]. Available: <http://www.unece.org/fileadmin/DAM/trans/conventn/crt1968e.pdf>
- [32] A. Rizaldi, J. Keinholz, M. Huber, J. Feldle, F. Immler, M. Althoff, E. Hilgendorf, and T. Nipkow, "Formalising and monitoring traffic rules for autonomous vehicles in Isabelle/HOL," in *integrated Formal Methods (iFM 2017)*, 2017, pp. 50–66.
- [33] S. Ovchinnikov, "Max-min representation of piecewise linear functions," *Contributions to Algebra and Geometry*, vol. 43, no. 1, pp. 297–302, 2002.