# Model Predictive Trajectory Planning for Automated Driving

Boliang Yi, Philipp Bender, Frank Bonarens, and Christoph Stiller

***Abstract*—In order to enable automated driving systems on the road, several key challenges need to be solved. One of these issues is real-time maneuver decision and trajectory planning. This paper introduces a general framework for maneuver and trajectory planning with model predictive methods. It discusses several representations of this framework in distinct complexity levels. In general, sophisticated models require to nonquadratic objective functions with nonlinear constraints, leading to increased computational complexities during calculation. Yet, oversimplified models can neither cope with vehicle dynamics in critical maneuvers nor do they represent complex traffic scenes with several maneuver options appropriately. A scheme to partition the trajectory space into homotopy regions is proposed. In each homotopy class, linearization about a trajectory from this class is applied. Demonstrations by simulation and with experimental vehicles show the capability of the proposed method in selecting optimal maneuvers and trajectories. This is even valid during extreme maneuvers, such as in last moment collision avoidance.**

***Index Terms*—Trajectory planning, maneuver decision, model predictive control, maneuver decoupling, homotopic classes.**

## I. Introduction

A FUNDAMENTAL skill of any autonomous mobile agent is the planning of appropriate motion. For automated road vehicles the trajectory planner must simultaneously meet requirements on various time scales. Foremost, in a short time horizon of a few seconds, it must guarantee collision-free trajectories with reasonable safety margin and comfort for all road participants. At the same time the planner will contribute to mid and long-term objectives related to high level modules like tactical decision-making and navigation. Numerous planning methods known from the robotics community aim at optimal motion with respect to a given objective criterion in restricted environments such as methods that find shortest path solutions in static scenes. Automated vehicles impose further challenges on motion planning due to their non-linear dynamics, the inherently imperfect and incomplete knowledge of the scene acquired by sensors and the unknown and reactive future motion behavior of other traffic participants. Furthermore, as compared to robotics, automated vehicles typically move at larger velocities and reachability of a safe state must be guaranteed at any time. The space of possible solutions is further restricted by real-time requirements.

In this work a general framework for maneuver and trajectory planning in the context of automated driving is presented. Model predictive methods are used to consider different levels of complexity for the planning problem. In addition, partitioning of the trajectory space with homotopic regions is proposed to find the optimal maneuver in scenarios with multiple obstacles. The capability of the proposed method is demonstrated by simulation and test vehicle experimentation.

### A. Related Work

This section compares different trajectory planning concepts for automated driving. Key assessment criteria shall be the capability for maneuver planning and execution in the context of automated driving. The concepts are segmented into trajectory planning concepts using geometrical primitives, potential field approaches and generic optimization criteria.

Geometric primitives have been used in many works to calculate trajectories with low computational effort. These primitives specify trajectories up to a few parameters and hence trajectory planning is reduced to parameter selection. A prominent, yet simple example is Dubins path, i.e. a sequence of circular arcs and straight lines that build the shortest curve connecting two vehicle poses in the two-dimensional Euclidean plane under limited curvature [1]. Circular arc trajectories have been investigated in the context of collision avoidance. The trajectories are composed of sequential circular arcs with continuous position and orientation. Naturally, due to the limitations in actuator and vehicle dynamics the discontinuous curvature can at most be followed approximately by real vehicles [2], [3].

Trajectories with maximum lateral displacement at a given longitudinal position for a point mass vehicle model with limited acceleration have been investigated in [4]. Such trajectories include a combination of braking and steering and can be considered to design an evasive trajectory to a single constant velocity obstacle.

Manuscript received June 27, 2017; revised February 10, 2018 and August 9, 2018; accepted September 11, 2018. Date of publication December 13, 2018; date of current version February 22, 2019. This work was supported by the German Federal Ministry for Economic Affairs and Energy (BMWi) in the frame of the third traffic research program of the German Bundestag under the research project Urbaner Raum: Benutzergerechte Assistenzsysteme und Netzmanagement. The work of C. Stiller was supported by the German Science Foundation DFG within the priority program Cooperatively Interacting Automobiles. *(Corresponding author: Boliang Yi.)*

B. Yi and P. Bender are with the Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany (e-mail: philipp.bender@kit.edu; boliang.yi@kit.edu).

F. Bonarens is with the Department of Advanced Technology, Electric and Electronic Systems, Opel Automobile GmbH 65423, Rüsselsheim, Germany (e-mail: frank.bonarens@opel.com).

C. Stiller is with the Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany, and also with FZI Forschungszentrum Informatik, 76131 Karlsruhe, Germany (e-mail: stiller@fzi.de).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIV.2018.2886683

Limitations of the build-up time of acceleration plays an essential role for maneuvers in critical situations. Sigmoidal equations have been proposed in [5], [6]. The authors show how the algorithm can be adapted to consider the build-up time of acceleration limited by actuators and vehicle dynamics.

Planning methods that parametrize geometric primitives have proven suitable in structured scenarios, such as for lane change and for simple collision avoidance maneuvers with a single object moving at constant velocity. However, they do not offer the flexibility to appropriately react under arbitrary environment conditions with multiple static and dynamic road participants.

Potential field approaches have been investigated in [7]–[9]. The basic idea is that environmental obstacles as well as road boundaries build a potential field which repels the trajectory of the vehicle. The vehicle shall then follow the direction of the negative gradient to prevent collisions with other obstacles and keep the vehicle near the center of the lane. Problems arise when overlapping influences of environmental information result in low gradients or local minima. Even though qualitatively, the influence of obstacles, road boundaries and moving traffic participants can intuitively be transferred to contributions to the potential field, their quantitative contribution is difficult to design. In consequence, potential field methods in general cannot provide guarantees for the calculated trajectory on requirements such as collision avoidance without any extension of the concept. In contrast, model predictive methods like the one proposed in this contribution can either guarantee absence of a collision in typical situation or at least allow to detect imminent collision well in advance.

Methods based on optimization criteria build on a generic framework in which relevant information about the environment and vehicle dynamics as well as desired driving properties can be properly formulated. The modules of the framework can be featured with elements of different complexity being suitable for the requirements of the considered maneuver. Optimal trajectory planning algorithms have been proposed by many researchers in different setups, which vary in module complexity as well as in computational burden [10]–[14]. Many approaches apply path-velocity decomposition to enforce convexity of the optimization criteria for the path and velocity profiles, respectively. A major restriction of these approaches to interactively optimize path and velocity is their requirement of appropriate initialization.

Simple setups of optimization-based methods which rely on a point mass model or kinematic model with a given start and end position but without inequality equations can be solved analytically without the need of complex solver algorithms [15]. However, the applicability is limited since environmental information such as obstacle position or geometry cannot be considered by inequalities in a way that allows for fast analytic solutions.

In complex scenarios with multiple objects and arbitrary road shapes general analytical solutions cannot be provided explicitly. A solution can be calculated by efficient numerical optimization solvers which are available for mathematically standardized formulations. A quadratic program is formulated with a linear model as well as a convex set to represent the environment in [11]. Test results are presented for automated driving over hundreds of kilometers on the Bertha Benz route in [16]. Similar works realized integrated trajectory planning and control using a model predictive control (MPC) method based on quadratic programming [13], [17], [18].

While these contributions have demonstrated the applicability of fixed-time solvers like quadratic programs for optimal trajectory planning in automated driving, these approaches are limited in the design of objective functions and constraints. Hence, their use is restricted to simple vehicle models and low complexity driving environments. Trajectory planning for challenging driving tasks such as collision avoidance in critical situations or selection of the best driving maneuver requires accounting for nonlinearities in the vehicle dynamics and in the environmental model.

A major source for non-convexity of objective functions in motion planning stems from the presence of static and dynamic obstacles. Therefore, several authors have separated maneuver planning from trajectory planning in a continuous space [19]. Through exploration of the state space for inevitable collision states (ICS) can provide real-time guarantees and the absence of collisions, yet does not guarantee local optimality [20]. Recently, quadratic programming has been extended to a mixed integer quadratic program (MIPQ), where integer variables additionally encode maneuver variables [21].

An explicit distinction of different maneuvers can also be achieved through the concept of homotopy classes. This concept is adopted in this contribution in order to partition the non-convex trajectory space into segments that allow for continuous optimization. Different homotopy classes basically arise due to presence of obstacles. Loosely speaking any two trajectories connecting the same start and goal states belong to the same homotopy class if they can be smoothly deformed into one another without traversing any obstacle. Hence homotopy classes can be used to distinguish *maneuvers* and have been proposed to express topological constraints for trajectory planning in robotics in two and three-dimensional configuration space [22], [23]. Different homotopy classes have been encoded in the integer variables of an MIPQ in [24]. So far, the homotopy concept has been applied to mostly static environments and to a low-dimensional state space. A major application has been the systematic environment exploration in robotics. Interestingly, a segmentation of the motion trajectory space by maneuvers with a completely different motivation has been proposed in [25]. The authors propose reinforcement learning for motion planning of self-driving cars within a rule-based safety concept termed *Responsibility-Sensitive Safety (RSS)* and argue that the search tree for possible motion increases prohibitively fast with planning time. In contrast, only few discrete maneuver options exist at any time and only few maneuvers can be sequenced in a typical planning horizon. Hence a maneuver planning tree would be substantially smaller and its complexity remains manageable.

Nonquadratic criteria with nonlinear constraints provide the opportunity to consider complex vehicle dynamics characteristics such as coupling effects between lateral and longitudinal dynamics [26], [27]. Still no efficient solvers are available for such a complex set of requirements, preventing real-time

deployment in vehicles. Sampling based methods fulfill real-time requirements by efficiently sampling the trajectory or command space [10], [28]–[30]. However, in general, hardly any guarantees on the resulting trajectories and their integrity can be provided.

Recent works approach this challenge with approximate optimal methods, leading to efficient optimizers. Successive linearization has been applied for vehicles in motor sport to control lateral dynamics and to track a given trajectory in [31]. By linearizing at each step around the operation point, the control method relies on a linear, yet time varying vehicle dynamics model, which can be used in quadratic programs efficiently.

The methods outlined so far maintain a state space with continuous values. Graph-based search methods may be applied by discretizing the state space. This allows for globally optimal solutions and implicitly avoids combinatorial ambiguities. A drawback of these methods is the high computational cost, which depends on the resolution of the discretized state space. Representative methods sample the space on regular lattices [32]–[34] or in a more randomized form, like rapidly exploring random trees which discover the space quickly [35]–[37].

### B. Contribution

This work provides a framework fulfilling key requirements for trajectory planning in the field of automated driving. An efficient framework should cope with the different complexity levels when planning trajectories for automated driving. The concept of homotopy classes is introduced as a scheme to partition the trajectory space into partitions. The resulting homotopy regions allow an efficient selection of the maneuver with the most optimal trajectory among several maneuvers.

Trajectory planning approaches to ensure drivability and integrity is provided while seeking for an efficient algorithm. The criterion drivability is fulfilled if the vehicle dynamic bounds are satisfied and if the implemented control system can realize the planned trajectory with the given actuator limitations. Integrity in the sense of this work comprises the absence of a collision with any obstacle when following the planned trajectory and the guarantee that the vehicle will not leave the drivable area limited e.g. by road boundaries or road markings. It can be concluded out of the introduction that the consideration of complex environmental information and appropriate vehicle dynamic models is essential when considering drivability and integrity of the planned trajectory. A key contribution shall be a framework that meets the demand for fundamental guarantees such as collision-freeness of the trajectory under reasonable assumptions. As mentioned, the algorithm should be computationally efficient for real-time application in a vehicle as the computational capacity representing a key cost driver is limited. After introducing an appropriate vehicle model and adding constraints for the environmental environment, both comprising non-linear equations, this inevitably leads to an additional step in the framework development. In this work, the algorithm is enhanced by using successive linearization of model and constraints.

### C. Outline

The outline of this paper is as follows. A mathematical formulation of trajectory planning is provided in Section II. Section III formulates planners in a general MPC framework and shows problem representations for different complexity levels. While general solvers that guarantee finding the global optimum in finite time do not exist for complex models, Section IV introduces the concept of homotopies for vehicle motion planning. Local linearization is proposed in Section V as a method that reduces computation time for the calculation of the local optima with sophisticated vehicle models in compliance with integrity constraints. Driving results in simulation and in an experimental vehicle are presented showing the applicability of the planning methods in real world driving.

## II. Problem Formulation

This section provides a problem statement and discusses various models and representations used in literature.

### A. Vehicle Dynamics

Vehicle and actuator dynamics are expressed in the discrete state space. At time instances $k = 0, 1, \ldots$ a command vector $\mathbf{u}(k)$ determines the state trajectory of the vehicle through a system model

$$\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k)). \tag{1}$$

The dynamic behavior $\mathbf{f}$ considers the kinetic effects of the command signal on the state vector $\mathbf{x}$ that comprises the degrees of freedom for the vehicle and its actuators. For simple models the state vector may only include the lateral and longitudinal position of the vehicle. More sophisticated vehicle models additionally involve the vehicle orientation as well as derivatives of position and orientation. For extreme driving situations, complex kinetic quantities like the spring damper extensions of each wheel may even be considered in the state vector.

The command vector $\mathbf{u}$ typically includes steering angle or its rate and acceleration or its corresponding jerk. Their magnitudes are typically restricted by vehicle design and by power consumption of its electrical and mechanical components. Examples for such limitations are the limits of the front wheel steering angle, the maximum steering torque or the maximum acceleration.

Eventually, the limitations on the command signal $\mathbf{u}$ can be separated in absolute bounds on $\mathbf{u}$ and on its change rate $\boldsymbol{\Delta}\mathbf{u}(k) = \mathbf{u}(k) - \mathbf{u}(k-1)$ defining a set of admissible command vectors

$$\mathcal{U} = \{\mathbf{u} \in \mathbb{R}^m \mid \mathbf{u} \in [\mathbf{u}_{\min}, \mathbf{u}_{\max}],\ \boldsymbol{\Delta}\mathbf{u} \in [\boldsymbol{\Delta}\mathbf{u}_{\min}, \boldsymbol{\Delta}\mathbf{u}_{\max}]\}. \tag{2}$$

Another set of drivability constraints refers to the state of the vehicle itself. These constraints guarantee stability of the vehicle and validity of the dynamic model. The validity of linear tire models, e.g. is restricted to low accelerations and stability may be lost for extreme state conditions [38]. Hence state constraints

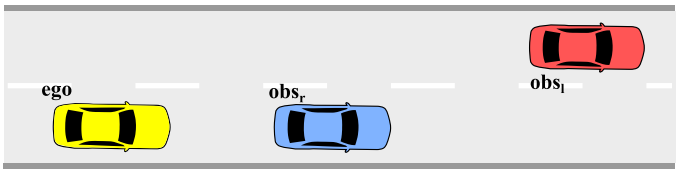



Fig. 1. Arbitrary traffic situations must be considered by the planning method. In the figure an ego vehicle and two obstacles $obs_\mathrm{r}$ and $obs_\mathrm{l}$ are illustrated.

are imposed that often restrict state variables to finite intervals

$$\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^n \,|\, \mathbf{x} \in [\mathbf{x}_\mathrm{min}, \mathbf{x}_\mathrm{max}]\}. \tag{3}$$

### B. Traffic Situation

Beyond considering the dynamics of the ego vehicle, the planning method must obviously consider the current traffic situation. As depicted in Fig. 1, a situation with the ego vehicle driving on a road in state $\mathbf{x}(k)$ in vicinity of $n > 0$ obstacles is considered.

*1) Static Integrity Constraints:* Static road infrastructure such as lane boundaries described by left and right markings, need to be considered by the planner. Their static characteristics are subsumed with static obstacles like vehicles parked on the roadside or other types of static traffic infrastructure. Defining an indicator function $h(\mathbf{x}, \mathbf{o}_{1,n})$ which is set to 1 if the vehicle in state $\mathbf{x}$ intrudes beyond the boundaries of the lane or static obstacles $\mathbf{o}_{1,n}$ and 0 otherwise, one yields the static integrity constraints

$$\mathcal{L}_i = \{\mathbf{x} \in \mathbb{R}^n \,|\, h(\mathbf{x}(i), \mathbf{o}_{1,n}) = 0\} \tag{4}$$

where $\mathbf{o}_{1,n}$ may comprise the positions, the lengths and the widths of the n obstacles. The constraints must be met throughout the planning horizon $i = k+1, \ldots, k+N$.

*2) Dynamic Integrity Constraints:* Moving traffic participants are particularly challenging to any motion planning method as they lead to an intense coupling of the constraints on states at different instances in time. Let $n$ moving traffic participants be located at position $\mathbf{x}^{(j)}(k)$, at time $k$, where each index $j = 1, \ldots, n$ refers to a particular moving object. Typically, their motions are at least known in the immediate past. For most planning purposes it is assumed that their future trajectories $\mathbf{x}^{(j)}(k+1), \ldots, \mathbf{x}^{(j)}(k+N)$, can be predicted for a planning horizon of $N$ time steps.[1] Typically, such a prediction assumes constant velocity or constant acceleration driving along the lanes of the road.

Then dynamic integrity constraints are imposed as

$$\mathcal{C}_i = \{\mathbf{x} \in \mathbb{R}^n \,|\, h(\mathbf{x}(i), C(\mathbf{x}^{(1)}(i), \ldots, \mathbf{x}^{(n)}(i))) = 0\}, \tag{5}$$

where the function $\mathcal{C}()$ generates the boundaries of all moving objects for their given states. The time-varying constraints must also be met throughout the planning horizon $i = k+1, \ldots, k+N$.

[1]Generally, prediction of future trajectories is subject to uncertainties. Nevertheless, the assumption of a certain prediction is used throughout the remainder of this paper for the sake of notational simplicity.

*3) Objective Function:* The hard constraints are defined so far to assure drivability and integrity of the planned trajectory. The set of trajectories meeting these constraints is large and thus additional constraints are required to resolve ambiguity. It may be argued, whether traffic rule compliance should be considered with hard or soft constraints. Further, the planner shall avoid unnecessary acceleration and calculate trajectories which are comfortable for the passengers. However, a beneficial balance of comfort, efficiency and safety margin to the hard constraints are typically quantified in the objective function. Frequently, a quadratic form

$$\begin{aligned} J(k) = &\sum_{i=0}^{N-1} \mathbf{u}(k+i)^T \mathbf{R}\mathbf{u}(k+i) \\ &+ \sum_{i=1}^{N-1} \mathbf{x}(k+i)^T \mathbf{Q}\mathbf{x}(k+i) \\ &+ \mathbf{x}(k+N)^T \mathbf{Q}_\mathrm{f}\mathbf{x}(k+N), \end{aligned} \tag{6}$$

is selected, where the positive semi-definite matrices $\mathbf{R}$, $\mathbf{Q}$ and $\mathbf{Q}_\mathrm{f}$ weight the costs of command vectors, state vectors and final state vector, respectively.

In summary, the goal of the motion planner at a given instance in time $k$ with state $\mathbf{x}(k) = \mathbf{x}_0$ is to determine a sequence of the command input

$$\tilde{\mathbf{u}} = \left(\mathbf{u}^T(k), \ldots, \mathbf{u}^T(k+N-1)\right)^T$$

that yields the best trajectory

$$\tilde{\mathbf{x}} = \left(\mathbf{x}^T(k+1), \ldots, \mathbf{x}^T(k+N)\right)^T$$

in the horizon of $N$ time steps in the sense of optimizing the objective function (6) under consideration of the drivability and the integrity constraints.

## III. MPC Trajectory Planning

In general, the solution of the trajectory planning problem as formulated in the last section cannot be provided for online application in the vehicle. Nevertheless, for careful selection of the individual models optimum trajectories can be generated with low computational effort or even analytically. Starting with simple vehicle models and traffic situations, which can be solved analytically, this section introduces more sophisticated vehicle models and realistic actuator constraints as well as traffic situations of larger complexity. An increase in complexity goes in parallel with computational burden. Based on these results, in the following sections, two approaches are shown where a nonlinear MPC algorithm can be approximated with a simple model that allows to find a good - yet not necessarily optimum - trajectory with fast techniques.

### A. Analytic Approach

Analytic solutions to trajectory planning exist in simple situations only and are designed for simple kinematic vehicle models. Nevertheless, they provide insight into expected trajectories and into the basic concept of the general structure shown later. To

apply the calculus of variation the objective function is adapted to account for continuous trajectories $\mathbf{x}(t) = (x(t), y(t))^T$ that describes the center position of the rear axle over continuous time $t$ in this section. Likewise, for continuous time the objective function is rewritten as

$$J(\mathbf{x}(t)) = \int_{t=0}^{t_N} L(t, \mathbf{x}(t), \dot{\mathbf{x}}(t), \ddot{\mathbf{x}}(t), \dddot{\mathbf{x}}(t))dt.$$

It is well known that the minimum of $J$ must satisfy the Euler-Poisson differential equation

$$\sum_{n=0}^{3}(-1)^n \frac{d^n}{dt^n}\frac{\partial L}{\partial \mathbf{x}^{(n)}} = \mathbf{0}.$$

Thus, for, e.g.

$$L(t, \mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}}, \dddot{\mathbf{x}}) = \|\dddot{\mathbf{x}}\|^2$$

one readily yields quintic decoupled polynomials for $x$ and $y$ minimizing the mean squared jerk [39]

$$\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \begin{pmatrix} \sum_{i=0}^{5} a_i t^i \\ \sum_{i=0}^{5} b_i t^i \end{pmatrix}.$$

The 12 parameters of such trajectories may be determined by the initial and final conditions. In the presence of additional constraints, the solution would remain a piecewise polynomial in all sections where the constraints are inactive.

### B. Quadratic Program

While analytic solutions provide insight about optimal trajectories in the absence of hard constraints under a mass point vehicle model, linear constraints and vehicle dynamics can be accounted for by quadratic programs [11], [13]. Path-velocity decomposition in combination with a linear vehicle model yields linear constraints for a wide range of traffic applications. This technique iterates between planning an optimum path for a given velocity profile and planning the optimum velocity for a given path. While this technique cannot strictly guarantee a global optimum trajectory, it has proven highly efficient in many applications.

Let us assume linear vehicle dynamics, such that eq. (1) can be rewritten as

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k)$$

this can be expanded to (cf. e.g. [13])

$$\tilde{\mathbf{x}} = \boldsymbol{\mathcal{A}}\mathbf{x}_0 + \boldsymbol{\mathcal{B}}\tilde{\mathbf{u}}$$

with

$$\boldsymbol{\mathcal{A}} = \begin{bmatrix} \mathbf{A}^T & (\mathbf{A}^2)^T & \cdots & (\mathbf{A}^N)^T \end{bmatrix}^T$$

$$\boldsymbol{\mathcal{B}} = \begin{bmatrix} \mathbf{B} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{AB} & \mathbf{B} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{A^{N-1}B} & \mathbf{A^{N-2}B} & \cdots & \mathbf{B} \end{bmatrix}$$

Insertion of this linear dynamics maintains the quadratic form of the objective function (6)

$$J(\tilde{\mathbf{u}}, \mathbf{x}_0) = \tilde{\mathbf{u}}^T\mathbf{H}\tilde{\mathbf{u}} + \mathbf{x}_0^T\mathbf{F}\tilde{\mathbf{u}} + \mathbf{x}_0^T\mathbf{G}\mathbf{x}_0 \tag{7}$$

with

$$\begin{aligned} \boldsymbol{\mathcal{Q}} &= \mathrm{diag}(\mathbf{Q}, \ldots, \mathbf{Q}, \mathbf{Q}_\mathrm{f}) \\ \mathbf{H} &= \mathrm{diag}(\mathbf{R}, \ldots, \mathbf{R}) + \boldsymbol{\mathcal{B}}^T\boldsymbol{\mathcal{Q}}\boldsymbol{\mathcal{B}} \\ \mathbf{F} &= 2\boldsymbol{\mathcal{A}}^T\boldsymbol{\mathcal{Q}}\boldsymbol{\mathcal{B}} \\ \mathbf{G} &= \boldsymbol{\mathcal{A}}^T\boldsymbol{\mathcal{Q}}\boldsymbol{\mathcal{A}}. \end{aligned}$$

Since the last term in (7) does not depend on $\tilde{\mathbf{u}}$, it can be ignored in the optimization process.

Linear constraints can be augmented to the objective function (7) without significantly increasing the computational burden required for optimization. These constraints may ensure drivability and integrity as outlined in Section II. As a simple example, for a given velocity profile, a road bottleneck for a point vehicle model would read

$$\mathcal{L}_i = \{\mathbf{x} \in \mathbb{R}^n \mid y(i) \in [y_{\min}, y_{\max}])\}, \tag{8}$$

where $y$ denotes lateral position and the constraint is imposed for all time instances $i$ when the vehicle is inside the bottleneck. Quadratic programs of this form cannot be solved analytically, but allow for a fast and fixed-time numerical optimization.

### C. General Objective Function

The technique outlined above can readily be extended to incorporate any linear dynamic model and linear drivability constraints as well as moving obstacles with known trajectory without affecting the efficiency of the optimization. However, any non-quadratic objective function, non-linear constraints, a non-linear vehicle model and the coupling of lateral and longitudinal planning will result in a more complex objective function that no longer can be solved with a quadratic program. The latter can be illustrated with a simple maneuver to pass a static obstacle. In general, the integrity constraints that prevent a collision remain in the form of eq. (8). However, the set of time instances $i$, when the constraint is imposed, depends on the velocity profile and hence a quadratic program may be trapped in local optima.

## IV. Maneuver Decoupling: Selection of Homotopic Classes

Solving quadratic and convex planning problems is possible with efficient algorithms. It has, however, been outlined that the coupling of longitudinal and lateral motion - while being crucial for effective planning in the presence of obstacles - leads to a problem that does not lead itself to a solution with a quadratic program.

In addition to that, in every day driving situations, there is usually *more than one* maneuver option to fulfill a driving task. And for each option, a human driver is likely to find an *optimal* solution, see Fig. 2. To illustrate this with a simple example: Consider following a slower car which exposes two options: staying behind or overtaking. In terms of this paper it means

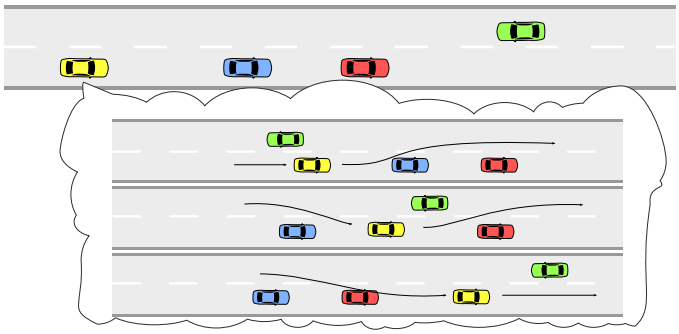


Fig. 2. An experienced driver of the yellow vehicle will identify at least three distinct maneuvers to overtake the preceding vehicles as depicted below. The proposed concept searches for an optimum trajectory for each maneuver first (these are the local optima) and then selects the *best* (global optimum) that avoids to crash into the oncoming green vehicle.

that trajectory planning has to cope with many *local* optima, and the planning module has to locate them and decide for the best option in the sense of a global optimum (limited to the detected local optima).

As discussed in Section I-A, graph-based search techniques have a strength in avoiding local optima in a discretized state space. Continuous methods, on the other hand, converge much faster and do not need the search space to be discretized, but get trapped in local optima. Therefore, decoupling of the *maneuver* decision from the trajectory planning is proposed in this work. While the set of maneuvers is discrete in nature and typically exhibits low cardinality, trajectory planning for a given maneuver is typically a task that may well be addressed by a continuous optimization method, such as quadratic programs. Loosely speaking, first the trajectory space is segmented into regions where at most a single local optimum is expected. Thus, each segment of the trajectory space lends itself to standard optimization procedures. Finally, the overall optimum motion is selected. Preliminary work on this idea has been published in [40].

In the remainder of this section an $x \times y$ coordinate system for spatial coordinates is used and it is assumed that the vehicle will travel along the $x$ coordinate and the road is straight. This way, the $x$ coordinate is identical with the longitudinal road coordinate, and $x \times t$ diagrams can be used to depict the corresponding longitudinal profile. The maneuver decision algorithm in this approach is not restricted to straight roads, but does not address the full contextual information like traffic rules and limitations on the perceived environment. This work assumes to dispose of sufficient sensor coverage, so each relevant object can be aware of. Occlusions are not considered.

### A. Homotopic Classes

In this section, a trajectory $\tilde{\mathbf{x}} : t \rightarrow (x, y)^\top \in \mathbb{R}^2$ is assumed to be a continuous curve in $\mathbb{R}^3$, and the $t$ component is monotonically increasing. Two trajectories $\tilde{\mathbf{x}}_1(t), \tilde{\mathbf{x}}_2(t)$ share a homotopic class relative to $F$ if there is a continuous function $h(\alpha, t)$ with $\alpha \in [0, 1]$ and $h(0, t) = \tilde{\mathbf{x}}_1(t)$ and $h(1, t) = \tilde{\mathbf{x}}_2(t)$ and $h(\hat{\alpha}, \hat{t})$ is a feasible state for each $\hat{\alpha}, \hat{t} \in \{[0, 1] \times [0, T]\}$. $F$ is a set of common elements of both trajectories. In the course of this paper, where the control input is planned until a certain point in time, this set comprises the first point in all three dimensions as well as the time coordinate of the last point. This means, that the last point is allowed to be moved spatially, but not in the temporal dimension.

A local planning method that works with gradients and tries to stay collision free will *not* exploit homotopic classes different than the one it has been initialized in – changing homotopic classes requires violation of constraints. In other words: in addition to other properties of local optimization methods, they will only find optima located in the homotopic class of the problem initialization. By adopting the concept of homotopies that has been proposed in mainly static robotics environments, e.g. [22], this work can thus explicitly address a major source for discontinuities in motion trajectory space and build continuous trajectory optimization schemes that can determine optimum trajectories even among different homotopy classes.

Instead, this behaviour is improved by identifying the different homotopic classes, finding an initialization for each one and solving the problem for each class. But there is a problem: the definition of homotopic classes is sound, but not directly applicable in the domain of trajectory planning with continuous methods. Making use of homotopy with continuous methods would mean that a proper initialization within each class has to be found, and then find a constraint function which makes sure that there will be an appropriate function $h(\cdot)$ between the initialization and every refinement step during the solution process. Unfortunately, there is no general way to obtain such a function. Instead, this work relies on another property of a continuous optimizer: by using only small steps, the optimizer will not be able to leave the homotopic class without violating geometric constraints. With this assumption (which may not hold true in every case, but in most), what is left to do is the *identification* of the distinct homotopic classes. This identification is something an experienced driver permanently does, Fig. 2 shows a scene where this assumption apparently holds true.

The key is the identification of homotopic classes in three dimensions (remember: the third dimension is *temporal*, not *spatial*). We can think of a solid volume which is colored black. Each point ($x$, $y$, $t$) in the volume is colored red if position ($x$, $y$) is occupied at time $t$. This way, the trajectories of individual traffic participants and geometric constraints (the shape of the road) can be included at the same time. The task is then to plan a trajectory using the black parts of the volume. A trajectory exclusively using this part of the volume is collision free. There are methods to explore the topology of a three-dimensional volume, for instance the medial axis transform [41] and homotopic thinning [42]. These methods yield graph-like, simplified representations of the planning volume. Each distinct path through these graphs corresponds to a distinct homotopic class. To use the paths as an initialization for a trajectory, two problems have to be solved. First, the initial solution has to be linked to the current position. Second, the optimization problem is formulated to optimize the *positions* for *given times*, so one component of the initialization needs to be evenly distributed along the temporal planning horizon. Both requirements are not considered in the mentioned methods, and apart from heuristics, there seems to be no natural solution to do so.

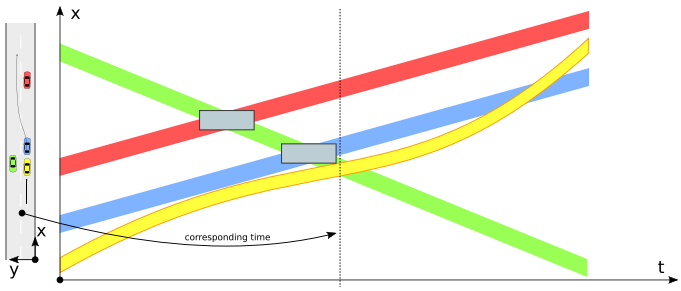



Fig. 3. The *critical sections* are marked in grey. In this example, the yellow vehicle decides to pass all critical sections *below*. The diagram shows the longitudinal road coordinate over time, the view on the left shows the situation at the time indicated by the dashed line.

The thought experiment showed that a purely geometric approach is appealing, but it lacks of a solid foundation to transfer the well-known methods from the topology domain into the domain of trajectory planning. So we go back to the abstraction level of *objects* to solve the problem of topology exploration. Fig. 2 indicates that it is the *interaction of objects* that determines the homotopic classes. There is a big difference whether the yellow vehicle overtakes blue *before* or *after* blue and green met. The key finding here is that for each pair of cars there is a conflict area in the (x $\times$ t) plane the ego vehicle has to strictly avoid. Those areas are marked with gray rectangles in Fig. 3. The road is blocked at a certain section (span of the rectangle in x direction) at a certain time (span of the rectangle in t direction), and the vehicle either has to be behind or in front of this section at this time.

Apart from these sections, the vehicle is allowed to cross other vehicles' traces in the $x \times t$ diagram: avoiding collisions is possible in the $y$ dimension.

### B. Constraints for Homotopic Classes

The above considerations may be generalized in the finding that a discrete decision arises for *each* critical section which may either be passed above or below. For $n$ vehicles, there may be up to $n-1$ sections leading to $2^{n-1}$ discrete decisions.

It is assumed that the other vehicles are independent of the motion of the ego vehicle. In the given example the decision to overtake the blue vehicle before meeting the green one implies overtaking the blue car before green and blue meet. This translates into the following bound constraints for the optimization task: Until meeting green, stay behind red; and after meeting green, move ahead of blue. For a straight road, those constraints would be real bound constraints on the state $\mathbf{x}$. If the road is not straight, but parameters are optimized in $(x \times y)$ space a curvilinear coordinate system needs to be introduced which is used to transform the optimization parameters, and evaluate the constraints there. For instance, the road may be represented as a centerline and a width. The centerline is given as a polygonal line. To transform a Cartesian point into the curvilinear coordinate frame, the corresponding segment of the polygonal line needs to be found, the point needs to be projected onto the segment and the distance can be evaluated, along with the arc length coordinate in the curvilinear system. With this method, the distance will be continuous, but discontinuities may arise in the gradient of the distance. Therefore, a continuous interpolation method like Phong shading, introduced to trajectory planning in [11] can be used. This increases the computation effort of a single iteration step by a constant computational effort since the trajectory has to be mapped and constraints as well as their gradients have to be evaluated in both curvilinear and Cartesian coordinates.

### C. Deriving and Combining the Constraints

Each pair of vehicles may produce a critical section. If this section is in the planning horizon it gives rise to the decision to pass above or below. Hence $n$ critical sections lead to $2^n$ possible decision sequences that is referred here as *maneuvers*. In each combination, each section restricts the longitudinal coordinate for an interval in time.

When a critical section spans over the interval $(x_{\text{low}}, x_{\text{high}}) \times (t_{\text{low}}, t_{\text{high}})$, the decision to pass below or above it constrains the trajectory to $(0, x_{\text{low}})$ for $t < t_{\text{high}}$ or $(x_{\text{high}}, \infty)$ for $t > t_{\text{low}}$, respectively. It is worth noting that constraints imposed by the individual decisions may conflict for some maneuvers. The specific maneuver is then infeasible and subsequent trajectory planning will be omitted. In real traffic situations the number of feasible maneuvers $N$ may be significantly below $2^n$.

### D. Scene Evolution and Noise: Robustness

Actions of the vehicle may lead other vehicles to adapt their maneuvers, and assumptions made during the homotopy identification may be incorrect. In each iteration of the planning step, the identification of the homotopic regions is executed. This makes the approach robust to such an evolution as long as the underlying model (about the motion of other vehicles) is reasonably correct in the time between two planning steps. However, in the worst case the selected maneuver may change in every planning step since the approach does not take recent decisions into account. In complex situations this may lead to behaviour which is hard to predict by the other vehicles. To dampen this effect, the homotopic region selected in a previous planning step may be preferred over a new candidate as long as the new candidate does not outperform the region by a certain degree. Since the construction of the homotopic regions is motivated by the other vehicles, the regions can be identified and tracked across several planning steps. The selection strategy is not in the scope of this paper.

Another effect of uncertainties in the observation of other objects is that the *existence* of homotopic classes could be treated as a probabilistic value. For instance, a gap between two vehicles may disappear if the observation of the vehicles is too unconfident. The presented approach cannot cope with these effects, the homotopic classes are treated as a discrete value.

### E. Maneuver Decoupling: Summary

The framework described above allows to partition the planning problem with moving obstacles into simpler ones, each

corresponding to a given maneuver. The framework allows to cope with arbitrarily shaped objects and roads and to get a locally optimal solution, since a local, continuous planner is used. As it is planned in two dimensions and not along a fixed path, maneuvers with varying paths may be executed, like overtaking maneuvers. For each subproblem, the complexity, properties and guarantees of the continuous planner apply. The combinatorial framework given here increases the time complexity exponentially, but many of the maneuver variants will be discarded in an early stage since constraints are conflicting. In the scenario depicted in Fig. 2, there may be four variants, but only three of them are non-conflicting. Furthermore, it may not be necessary to consider all maneuvers at a given time, e.g. when the velocity on the own lane is sufficient, lane change maneuvers to faster lanes may be discarded.

## V. Planning Maneuvers for Combined Longitudinal and Lateral Dynamics

In the previous section, the trajectory space has been partitioned into regions representing distinct maneuvers defined by their homotopic class. While simple models and linear dynamics suffice for trajectory planning in most traffic situations, more sophisticated models are required for critical maneuvers that approach the handling limits of the vehicle. This section presents a model predictive planner in the context of evasive collision avoidance maneuvers. Starting with several nonlinear constraints, approximate solutions will be discussed subsequently. By local linearization of nonlinearities, linear time varying quadratic programs are reformulated and demonstrated in simulation and real time application. To clarify the essential points, this paper will focus on two inequality constraints only, the vehicle dynamics model and the dynamic integrity constraint. Further constraints will be mentioned but not explained in detail for brevity.

### *A. Nonlinear Optimal Trajectory Planning for Collision Avoidance Maneuvers*

The nonlinear optimal planning method follows the structure discussed in Section II. The modules of the general framework are featured with nonlinear components for each module, to match requirements resulting from collision avoidance tasks, i.e. nonlinear vehicle dynamics at dynamic limits and position predictions to avoid collision.

*Vehicle Dynamics Model:* A nonlinear vehicle dynamics model integrating the longitudinal as well as the lateral dynamics is used. The model considers the lateral forces with tire models proposed by Pacejka [43] and considers the tire load change due to longitudinal acceleration. The lateral dynamics is represented by the single-track model. The state vector $\mathbf{x} = (\beta, \dot{\varphi}, \varphi, y, \delta, \dot{\delta}, x, v, a_x)^T$ comprises the side slip angle $\beta$, the yaw rate $\dot{\varphi}$, the yaw angle $\varphi$, the positions $x, y$, the steering angle $\delta$, the steering rate $\dot{\delta}$, the longitudinal velocity $v$ and the longitudinal acceleration $a_x$. The command signal $\mathbf{u} = [T, a_{x,\text{cmd}}]^T$ includes steering torque and commanded longitudinal acceleration. The continuous differential equation is given as

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$$

$$\mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) = \begin{bmatrix} (F_{yf}(t) + F_{yr}(t))/(mv(t)) - \dot{\varphi}(t) \\ (F_{yf}(t) l_f - F_{yr}(t) l_r)/J_z \\ \dot{\varphi}(t) \\ v(t)\sin(\varphi(t) + \beta(t)) \\ \dot{\delta} \\ (T - 2D_{Str}\alpha_f - d_{Str}\dot{\delta})/J_{Str} \\ v(t)\cos(\varphi(t) + \beta(t)) \\ a_x(t) \\ -\frac{1}{t_{\text{del}}} a_x(t) + \frac{1}{t_{\text{del}}} a_{x,\text{cmd}}(t) \end{bmatrix} \tag{9}$$

$$F_{yf}(t) = m \cdot \left( \frac{l_r}{l} \cdot g - \frac{h_c}{l} a_x(t) \right) \cdot \mu_f$$

$$F_{yr}(t) = m \cdot \left( \frac{l_f}{l} \cdot g + \frac{h_c}{l} a_x(t) \right) \cdot \mu_r$$

$$\mu_f = \text{tire}(\alpha_f(t))$$

$$\mu_r = \text{tire}(\alpha_r(t))$$

$$J_{Str}\ddot{\delta} + d_{Str}\dot{\delta} = T - 2D_{Str}\alpha_f(t)$$

$$\text{tire}(\alpha(t)) = D\sin(C\arctan(B\alpha(t) + E(B\alpha(t) - \arctan B))).$$

Here $m$ is the mass of the vehicle, $J_z$ the yaw inertia, $l_f$ and $l_r$ are the distances of the center of gravity to the front and rear axis respectively. $F_{yf}$ and $F_{yr}$ are the lateral forces generated by the tires at the front and rear axis. $F_{zf}$ and $F_{zr}$ are the wheel loads of the front and rear tires respectively. Pacejka's tire formula [43] with $\alpha_f$ and $\alpha_r$ as the slip angles for the front and rear tires is used for the tire function. $\delta$ is the steering wheel angle, $\varphi$ is the yaw rate, $\beta$ is the side slip angle and $a_x$ is the longitudinal acceleration of the vehicle. $t_{\text{del}}$ is the time constant describing the longitudinal $\text{PT}_1$ characteristic. $J_{Str}$ is the inertia, $d_{Str}$ the damping and $D_{Str}$ the self alignment parameter of the steering system.

The command signals in $\mathbf{u}$ are limited by the admissible actuator range $\mathcal{U}$ considering the absolute bounds as well as the change rate limits of the steering torque and the commanded acceleration.

$$\mathcal{U} = \{\mathbf{u} \in \mathbb{R}^2 \mid -T_{\max} \le T \le T_{\max},\, a_{\text{cmd,lim}} \le a_{x,\text{cmd}} \le 0, \\ -T_\Delta \le \Delta T \le T_\Delta,\, -a_{\text{cmd},\Delta} \le a_{x,\text{cmd}} \le a_{\text{cmd},\Delta}\} \tag{10}$$

The state contraints in $\mathcal{V}$ consider limitations in velocity and acceleration dynamics due to the friction circle. The vehicle's velocity is constrained to positive values since no backward driving can be induced by braking intervention. Further, the applicable forces for the vehicle dynamics are limited by the friction circle. Considering the friction circle on each tire is however not possible due to the characteristic of the single-track model. Instead, the acceleration circle represents the friction

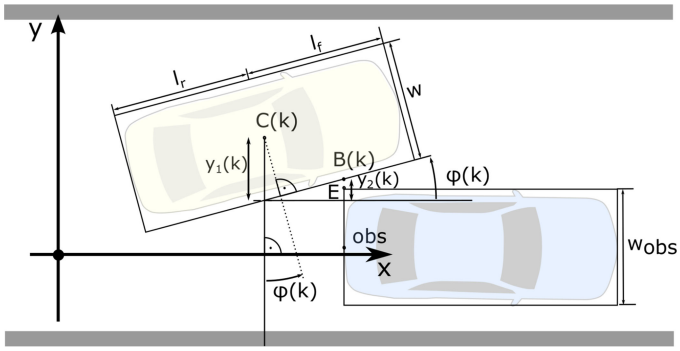



Fig. 4. Situation for evasive collision avoidance maneuvers of an obstacle on the right - Illustration of characteristic points, lengths and angles.

circle in the g-g diagram from [44]. These state constraints thus yield

$$\mathcal{V} = \{\mathbf{x} \in \mathbb{R}^9 \,|\, v \geq 0,\ \sqrt{{a_x}^2 + a_y^2} \leq \mu g,\ a_x \leq 0\}\ . \quad (11)$$

In addition, the control algorithm shall ensure stability when controlling the vehicle at high dynamics. Here boundaries are chosen for the yaw rate and the side slip angle $\dot{\varphi}_{\max}$ and $\dot{\beta}_{\max}$ to reflect the unstable vehicle states as described in [45]. The admissible set for the stability is thus given by

$$\begin{aligned}\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^9 \,|\, &-\dot{\varphi}_{\max} \leq \dot{\varphi} \leq \dot{\varphi}_{\max}, \\ &-\beta_{\max} \leq \beta \leq \beta_{\max}\}. \end{aligned} \quad (12)$$

*Dynamic Integrity Constraints:* One of the main objectives in this work is collision avoidance of obstacles realized by dynamic integrity constraints. We search for trajectories where the position of the vehicle avoids collision with obstacles in any traffic scenario like the one given in Fig. 1. This set of states depends nonlinearly on the longitudinal position, lateral position and yaw angle as well as on the geometry of the vehicle in the function $h(\mathbf{x}(i), C(\mathbf{x}^{(j)}(i)))$ (see eq. 5). As illustrated in Fig. 4 for rectangular object shapes with the length $l_{obs}$ and the width $w_{obs}$ the constraints read

$$\begin{aligned}\mathcal{C} = \{\mathbf{x} \in \mathbb{R}^9 | &h_1(y,\varphi) \geq y_{\text{obsl}}, \forall x \in \mathcal{X}_{or}, \\ &y_{\text{obsr}} \geq h_2(y,\varphi), \forall x \in \mathcal{X}_{ol}\}, \end{aligned} \quad (13)$$

$$\begin{aligned}\mathcal{X}_{or} := \{x \in \mathbb{R} | &(x_{fr}(k) > x_{\text{obsr}}) \cap \\ &(x_{rr}(k) < x_{\text{obsr}} + l_{\text{obsr}})\} \end{aligned} \quad (14)$$

$$\begin{aligned}\mathcal{X}_{ol} := \{x \in \mathbb{R} | &(x_{fl}(k) > x_{\text{obsl}}) \cap \\ &(x_{rl}(k) < x_{\text{obsl}} + l_{\text{obsl}})\} \end{aligned} \quad (15)$$

where $y_{\text{obsr}}$, $y_{\text{obsl}}$, $x_{\text{obsr}}$ and $x_{\text{obsl}}$ describe the lateral and longitudinal position of the obstacle on the right or left side of the vehicle and $x_{fr}$, $x_{rr}$, $x_{fl}$ as well as $x_{rl}$ are the longitudinal positions of the vehicle corners. In Fig. 4 the situation is illustrated for an obstacle on the right lane. The center of the obstacle edge obs and the vehicle position $x, y$ and orientation $\varphi$ at time $k$ is shown in the picture. $x_{\text{obsr}}$ and $y_{\text{obsr}}$ denote the longitudinal and lateral positions of the relevant obstacle edge.

Note that planning takes advantage of the homotopy concept introduced in Section IV by considering only the relevant edge of the obstacle given by the homotopy class of the maneuver. Further, the collision avoidance constraint presented above is active for the relevant side only when the obstacle edge lies between the longitudinal range of the ego vehicle.

*Objective Function:* The objective function serves to design the desired performance of the planning approach. Typically, a balance between comfort, efficiency and safety margin is considered. One of the most challenging trajectory design tasks is to plan evasion trajectories for collision avoidance in critical situation with short collision times. One design goal in (16) is to reduce yaw angle with a weight $q_\varphi$ in order to yield trajectories steering the vehicle back to the original orientation. Furthermore, this work aims to reduce velocity with a weight $q_v$ to minimize collision energy in case of an unavoidable impact. Finally, command input signals shall be minimized with the weights $r_a$ and $r_T$.

$$\begin{aligned}J(k) = &\sum_{i=1}^{N} \left[q_\varphi \varphi^2(k+i) + q_v v^2(k+i)\right] \\ &+ \sum_{i=0}^{N-1} \left[r_T T^2(k+i) + r_a a_x^2(k+i)\right]. \end{aligned} \quad (16)$$

*Optimization Problem:* The overall nonlinear optimization problem considers the presented vehicle dynamics model, admissible actuator range, state constraints, stability constraints, dynamic integrity constraints and the objective function. Furthermore, static integrity constraints $\mathcal{L}$ are used to consider lane boundaries, which can be realized by a simple inequality constraint on the lateral position.

$$\begin{aligned}&\min_{\tilde{\mathbf{u}}}\ J(k) \\ &\text{s.t. } \mathbf{x}(k+i+1) = \mathbf{x}(k+i) + t_d \cdot \mathbf{f}(\mathbf{x}(k+i), \mathbf{u}(k+i)) \\ &\quad \mathbf{u}(k+i) \in \mathcal{U} \\ &\quad \mathbf{x}(k+i) \in \mathcal{V} \cap \mathcal{S} \cap \mathcal{L} \cap \mathcal{C} \\ &\quad \forall i = 1, \ldots, N. \end{aligned} \quad (17)$$

The above equation describes an optimization problem with a cost function which is non-quadratic in the command input and contains nonlinear constraints. In the sequel, local approximations will be introduced that transfer the formulation to a quadratic cost function with linear constraints. Since the approximation depends on the operating point, the solver may be applied iteratively. A simpler approach is to take the trajectory from the last time step as initialization. This strategy is well suited to the homotopy concept as it avoids local optima within each class.

### *B. Approximate Optimal Trajectory Planning for Collision Avoidance Maneuvers*

The constrained optimization task given by eq. (17) is approximated in order to allow for efficient solvers. Specifically,

the combined vehicle dynamics model is linearized around predicted state trajectories and reformulate the collision avoidance constraint without explicit longitudinal position, such that the constraint is convex.

*Successive Linearization of Vehicle Dynamics Model:* To reduce model complexity, the vehicle dynamics (longitudinal and lateral) will be simplified in this section to a linear time varying model. This can be achieved by successive linearization around operating points of the predicted trajectory from the last control step. The approach assumes a high control sampling rate such that the predicted velocity between two consecutive control steps does not change significantly. The continuous state space equation is based on the vehicle dynamics model from [6] and [46]. The equation for a single prediction step $i$ is given below, where the index i is omitted for better readability.

$$\dot{\mathbf{x}}(k) = \mathbf{A}(k) \cdot \mathbf{x}(k) + \mathbf{B}(k) \cdot \mathbf{u}(k) + \mathbf{E}(k)$$

with $\mathbf{A}(k) =$

$$\begin{bmatrix} A_1 & A_2 & 0 & 0 & 0 & A_3 & 0 & A_4 & 0 \\ A_5 & A_6 & 0 & 0 & 0 & A_7 & 0 & A_8 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_{k-1} & 0 & v_{k-1} & 0 & 0 & 0 & 0 & 0 & 0 \\ A_9 & A_{10} & 0 & 0 & A_{11} & A_{12} & 0 & A_{13} & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{t_{\text{del}}} \end{bmatrix}$$

$$A_1 = -\frac{c_{\alpha_f,k-1} + c_{\alpha_r,k-1}}{m \cdot v_{k-1}};$$

$$A_2 = \frac{c_{\alpha_f,k-1} \cdot l_r - c_{\alpha_r,k-1} \cdot l_f}{m \cdot v_{k-1}^2} - 1;$$

$$A_5 = \frac{c_{\alpha_f,k-1} \cdot l_r - c_{\alpha_r,k-1} \cdot l_f}{J_z};$$

$$A_6 = \frac{c_{\alpha_f,k-1} \cdot l_r^2 - c_{\alpha_r,k-1} \cdot l_f^2}{J_z \cdot v_{k-1}};$$

$$A_3 = \frac{c_{\alpha_f,k-1}}{m v_{k-1} i_{\text{L}}};\ A_7 = -\frac{c_{\alpha_r,k-1} l_r^2 + c_{\alpha_f,k-1} l_f^2}{J_z v_{k-1}};$$

$$A_4 = \frac{c_{\alpha_{f,k-1}} + c_{\alpha_{r,k-1}}}{m v_{k-1}^2}\beta_{k-1} - 2\frac{c_{\alpha_{r,k-1}} l_r - c_{\alpha_{f,k-1}} l_f}{m v_{k-1}^3}\dot{\varphi}_{k-1} \cdots - \frac{c_{\alpha_{f,k-1}}}{i_L m v_{k-1}^2}\delta_{k-1};$$

$$A_8 = \frac{c_{\alpha_{f,k-1}} l_f^2 + c_{\alpha_{r,k-1}} l_r^2}{J v_{k-1}^2}\dot{\varphi}_{k-1};\ A_9 = \frac{2D_{Str}}{J_{Str}};$$

$$A_{10} = \frac{2 l_f D_{Str}}{v_{k-1} J_{Str}};\ A_{11} = -\frac{-d_{Str}}{J_{Str}};\ A_{12} = \frac{-2D_{Str}}{i_L I_{Str}};$$

$$A_{13} = \frac{-2 l_f D_{str}}{J_{str} v_{k-1}^2}\dot{\varphi}_{k-1};$$

$$\mathbf{B}(k) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{J_{Str}} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \frac{1}{t_{\text{del}}} \end{bmatrix} \mathbf{E}_k = \begin{bmatrix} \frac{F_{f0,k-1} + F_{r0,k-1}}{m v_{k-1}} \\ \frac{F_{f0,k-1} l_f + F_{r0,k-1} l_r}{J_z} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{18}$$

The state space matrices above are adapted in each control step, accounting for changing parameters on predicted operation points of the previous control step. The parameters $v_{k-1}$, $c_{\alpha_f,k-1}$, $c_{\alpha_r,k-1}$, $F_{f0,k-1}$ and $F_{r0,k-1}$ are predicted velocities, cornering stiffness parameters as well as constant terms in the linearized tire equations [46] for the predicted slip angles $\alpha_{fp}(k-1)$ and $\alpha_{rp}(k-1)$ at the corresponding prediction step $i$ of the last trajectory. The discrete state space matrices $\mathbf{A}_d$, $\mathbf{B}_d$ and $\mathbf{E}_d$ can be obtained by discretization with the control sampling time $t_d$ by using the matrix exponentials.

The nonlinear constraint corresponding to the acceleration circle in (11) has been approximated in [18] by several half spaces using linear constraints. Since a braking actuator can only cause negative longitudinal accelerations, six linear equations are used to approximate the left half space of the diagram. The modified set for state constraints can be given by

$$\mathcal{V}_q = \{\mathbf{x} \in \mathbb{R}^9 \mid v \geq 0,\ \mathbf{L}_y a_y + \mathbf{L}_x a_x \leq \mathbf{M},\ a_x \leq 0\}. \tag{19}$$

For further details on the chosen approximation please refer to [18].

*Approximation of Dynamic Integrity Constraints:* Dynamic integrity for collision avoidance has been formulated as nonlinear inequality constraints in (13)–(15). Since collision avoidance depends on both the lateral and the longitudinal position, the corresponding dynamics are planned together. To solve the problem as a quadratic program, linear inequalities must be formulated for the corresponding time steps. In this section, restrictions on the lateral position and yaw angle for each prediction time within the horizon are set. As outlined before simultaneous lateral and longitudinal planning result in constraints that depend on velocity. Following the principles of successive linearization, the constraints are imposed on the trajectory from the previous control step

$$\begin{gathered} y_{B_R}(k) > y_{\text{obs},r} + w_{\text{obs}}/2 \\ y_{B_R}(k) = y(k) + \Delta x \varphi - w/2 \\ \forall\ (x_{fr,k-1} > x_{\text{obs},r}) \cap (x_{rr,k-1} < x_{\text{obs},r} + l_{\text{obs}}) \\ y_{B_L}(k) < y_{\text{obs},l} - w_{\text{obs}}/2 \\ y_{B_L}(k) = y(k) + \Delta x \varphi + w/2 \\ \forall\ (x_{fr,k-1} > x_{\text{obs},l}) \cap (x_{rr,k-1} < x_{\text{obs},l} + l_{\text{obs}}) \\ \text{with}\ \ k = 1, \ldots, N,\ n = 1, \ldots, N_{obj}\,. \end{gathered}$$

Note that $x_{fr,k-1}$, $x_{rr,k-1}$, $x_{\text{obs},k-1}$ and $y_{k-1}$ are known prior to the actual optimization step, such that the dependency on the

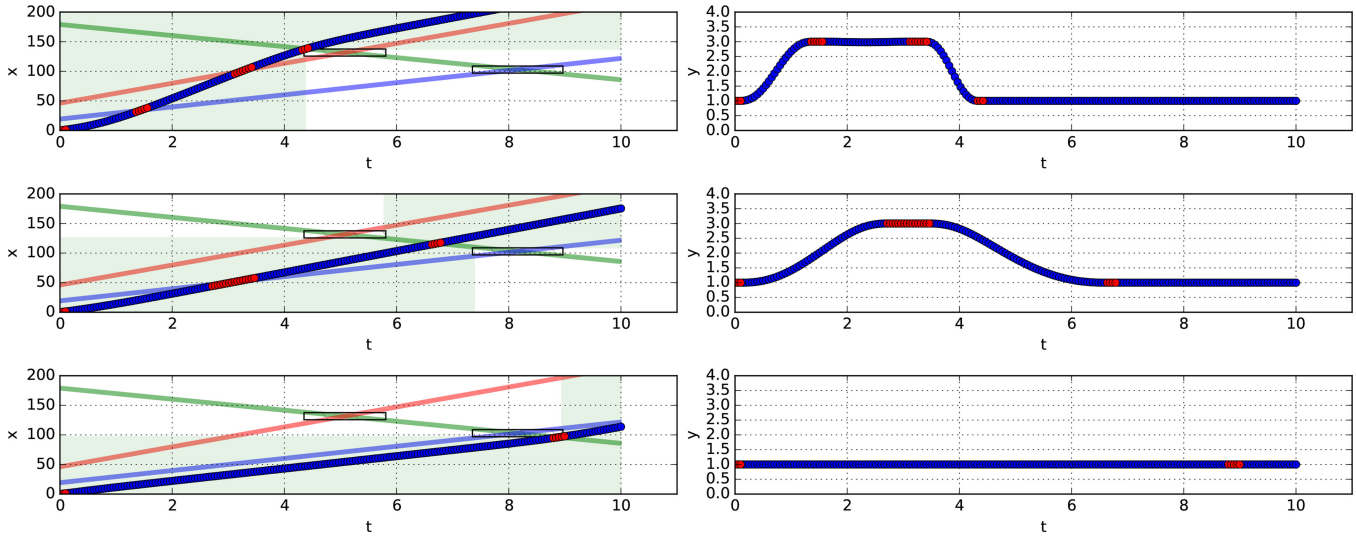



Fig. 5. Separation of homotopic classes. In the shown scenario, there are three distinct variants. Each line shows such a variant. The left-hand column shows the longitudinal coordinate over time, the right-hand column shows the lateral component. The points marked in red have been pinned after planning the longitudinal profile to prevent collision and meet the expectancies of the other road users. The green area is the feasible area allowed by the bound constraints on the longitudinal coordinate.

longitudinal position is eliminated in the optimization problem and the inequality constraint becomes linear.

*Optimization Problem:* The approximate optimization problem with the modified linearized vehicle dynamics model, as well as the linearized dynamic and static integrity constraints finally becomes

$$\begin{aligned} &\min_{\tilde{\mathbf{u}}} \; J(k) \\ \text{s.t. } &\mathbf{x}(k+i+1) = \mathbf{A}_d(k+i)\mathbf{x}(k+i) \\ &\quad + \mathbf{B}_d(k+i)\mathbf{u}(k+i) + \mathbf{E}_d(k+i) \\ &\mathbf{u}(k+i) \in \mathcal{U} \\ &\mathbf{x}(k+i) \in \mathcal{V}_q \cap \mathcal{S} \cap \mathcal{L} \cap \mathcal{C}_q \\ &\forall i = 0, \ldots, N. \end{aligned} \tag{20}$$

Note that the optimization problem consists of the same objective function, actuator limitations, stability boundaries and static integrity constraints as in problem (17).

## VI. Results

Section IV and V presented a trajectory planning framework for autonomous driving where the global planning problem is resolved using the selection of homotopic classes and successive linearization. The complete framework has been implemented in a simulation environment as well as in a real vehicle. The full functionality will be demonstrated using selected scenarios in this section. The performance of the maneuver decision method will be shown in Subsection VI-A by a complex scenario representing an overtaking maneuver typical for highway traffic, considering a high planning horizon. In Subsection IV-B the trajectory planning method is demonstrated in collision avoidance situations requiring maneuvers which can be considered as a response to disturbance rather than a planned overtaking maneuver. The proposed framework is covering both aspects of automated driving. The prediction horizon for this maneuver has been limited to 1.5 s for the sake of calculation effort. The planning horizon for the evasive maneuver not being limited by the method may be adapted.

### A. Maneuver Decoupling

First, the scenario shown in Fig. 2 will be used for a simulative experiment of the space partitioning scheme described in Section IV. For further simplification the path-velocity decomposition has been applied. The optimization is divided in three steps:

1) Identification of maneuver variants,
2) optimization of the longitudinal profile for each variant and
3) optimization of the lateral profile for each longitudinal profile.

Fig. 5 shows the result for the given example. The result indicates that there are at least three local minima. However, it can *not* be predicted which minimum the optimizer had chosen *without* the maneuver decoupling. Standard solvers would produce a trajectory that strongly depends on the initialization. The decoupling of longitudinal and lateral components needs to be explained a little bit further. On the left side, the result of the longitudinal optimizations for each variant can be seen. The black box marks the critical sections the vehicle has to *strictly* avoid. Of course, the vehicle is allowed to overtake by using the lateral space. This needs to be done whenever the longitudinal profile of the ego vehicle intersects the path of another vehicle. These points can easily be identified and are marked in red in the plot. The red points are laterally set to the overtaking lane and kept fixed during the optimization process. This way, a natural behavior during overtaking is achieved. The result in the $x \times y$

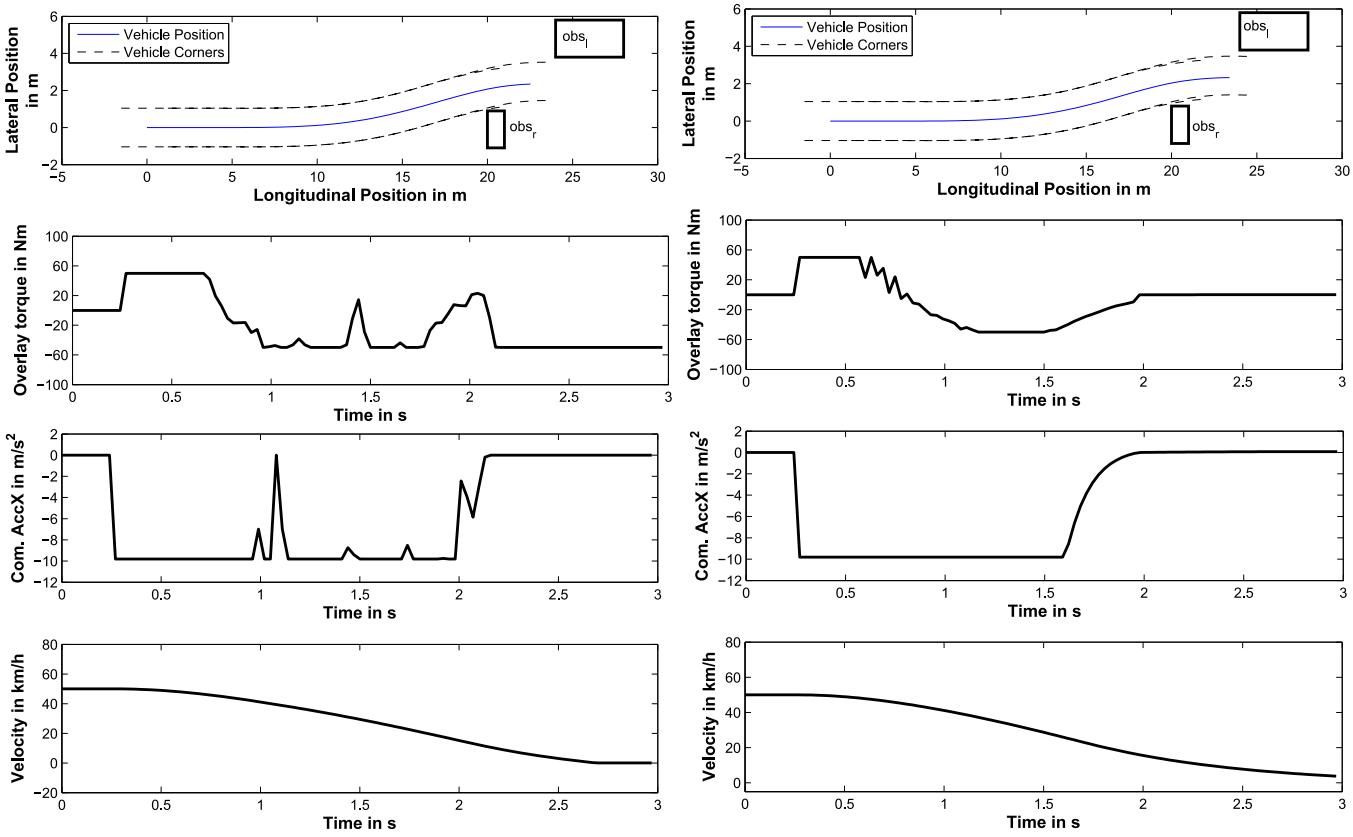



Fig. 6. Collision avoidance in a scenario with multiple obstacles. The left-hand column shows the maneuver performance with the nonlinear planning problem in Subsection V-A, the right-hand column shows the maneuver performance with approximated optimization problem presented in Subsection V-A. In the first row, the trajectory of the center of gravity (solid line) and of the vehicle corners (dashed lines) are shown. The commanded torque sequence, the commanded deceleration and the resulting velocity are depicted in the following rows.

domain is shown in the right-hand plots using the same color scheme.

### *B. Planning Maneuvers With Combined Dynamics*

In the second stage of the framework, the approximate trajectory planning method has been introduced which can be applied to plan a continuous trajectory for the maneuver preselected by the maneuver decision algorithm. In the scenario visualized in Fig. 1, the ego vehicle is driving with 50 km/h on the right lane of a straight road and needs to avoid collision with static obstacles on the lane. The first static obstacle on the right lane is located 20 m from the center of gravity of the ego vehicle. The distance is chosen such that the vehicle cannot brake to standstill without a collision with the obstacle. Therefore, an evasive collision avoidance maneuver must be accomplished to avoid collision with the obstacle. After that, collision avoidance with a second obstacle must be considered, which is located 24 m ahead of the initial position of the ego vehicle on the left lane. The scenario is chosen such that the ego vehicle can drive an intermediate course between both obstacles.

The algorithms for both planning methods have been implemented in simulation with a sampling rate of $dt_{\text{ctrl}} = 0.03$ s, a prediction horizon of 1.5 s, a prediction sampling time of $dt_{\text{pred}} = 0.1$ s and $\text{N} = 15$ prediction steps. The algorithms are running on a computer with Intel(R) Core(TM) i5-3380M, 2.9 GHz, 8 GB RAM and Windows 7 SP1. The planning parameters are summarized in Table I. The results in Fig. 6 are devoted to the performance of the planning method due to successive linearization compared to a nonlinear method in a simulative environment.

TABLE I
RELEVANT PARAMETERS

| Parameters | Values |
|---|---|
| $m$, $J_z$, $l_f$, $l_r$ | 2050 Nm, 3500 kg/m, 1.23 m, 1.51 m |
| $J_{Str}$, $d_{Str}$, $D_{Str}$ | 1.32 kg/m, 2.54 Nms/rad, 460 Nm/rad |
| $B$, $C$, $D$, $E$ | 2.1, 0.4, 19.6, $-0.7$ |
| $q_{\text{s}}$, $q_{\varphi}$, $q_{\text{v}}$ | $10^6$ 1/m$^2$, $3{\cdot}10^3$ 1/rad$^2$, 10 s$^2$/m$^2$ |
| $r_{\text{T}}$, $r_{\text{a}_\text{x}}$ | 0.1 1/Nm$^2$, 0.1 s$^4$/m$^2$ |
| $T_{\max}$, $T_{\Delta}$ | 50 Nm, 2000 Nm/s |
| $a_{\text{x,cmd,max}}$, $a_{cmd,\Delta}$ | -9.81 m/s$^2$ , 30 m/s$^3$ |
| $\dot{\phi}_{\max}$, $\beta_{\max}$ | 0.71 rad/s, 0.18 rad |
| N, $dt_{\text{pred}}$, $dt_{\text{ctrl}}$, $l_{obs}$, $w_{obs}$ | 15, 0.1 s, 0.03 s, 4 m, 2 m |

Fig. 6 indicates that both algorithms are able to avoid collision with the obstacles. For the purpose of discussing the results, the steering intervention is split into two phases, namely the steering phase and the counter steering phase. In the steering phase maximum torque is applied to avoid the collision with the first obstacle. In the second phase counter steering with negative

TABLE II
METRIC VALUES OF PLANNING METHODS IN SIMULATION

| Scenarios | Velocity Reduction | Final Yaw Angle | Computation Time |
|---|---|---|---|
| | $\Delta v_{\text{red}}$ | $\varphi_{\text{fnl}}$ | $t_{\text{cmp}}$ |
| Nonlinear Planner | 44.2% | $6^\circ$ | $10s - 15s$ |
| Approximate Planner | 45% | $1^\circ$ | $0.15s$ |
| Real Vehicle | 28.2% | $-9^\circ$ | $0.13s$ |

torque is applied to steer the vehicle back and avoid collision with the second obstacle. With respect to the longitudinal dynamics, the algorithm applies full braking first and reduces intervention until the vehicle stops. Further, the figure indicates that the performance of the nonlinear planning method shows short dips in the steering and braking intervention which did not show up for the approximated planning method on the right column. In addition, the performance of the nonlinear planning method shows a clear stabilization phase where the steering torque reaches 20 N at the end of the maneuver.

Table II shows the selected metric values of the planning methods. In this section, the metric values are chosen to be the velocity reduction to the time when the front of the ego vehicle is at the same longitudinal position as the obstacle and the final yaw angle when the vehicle is at standstill. Further the computation time per optimization step is compared to demonstrate the effect of successive linearization. The table shows similar performance metric values for both the nonlinear planner and the approximate planning method in terms of velocity reduction and yaw angle, while a significant reduction of computation time for the approximate method was achieved.

In the experiments, the computation time per optimization step takes approximately 0.13 s on the dSpace MicroAutobox II in the Opel Insignia experimental vehicle. The vehicle is equipped with an electrical power steering and an electronic brake control module out of serial production with software adaptations. This enables to command an additional steering torque interface of up to 100 Nm and a longitudinal deceleration limit of $-10$ m/$s^2$. The algorithm has a sampling rate of $dt_{\text{ctrl}} = 0.13$ s, a prediction horizon of 1.5 s, a prediction sampling time of $dt_{\text{pred}} = 0.15$ s and N $= 10$ prediction steps. The sampling rate deviates from the simulation environment due to the computational limitations of the processor in the MicroAutobox II which could be improved by the use of more advanced processors. The vehicle state information is partly available due to sensor signal conversion from IMU, wheel speed as well as steering angle sensor and partly based on kalman filter estimation.

Fig. 7 shows the performance of the algorithm in the vehicle. The objective of this test is to demonstrate the applicability of the proposed method. The scenario was designed to bring the vehicle at the limits of sliding regions to investigate the performance of the algorithm in acceleration areas above 8 meter per second square. When comparing simulation with testing results, it must be taken into account that ABS and ESP impact has not been modelled for sake of effort reduction. Still the comparison clearly shows the similarity of the different phases with steering and counter steering. In the test vehicle a stabilizing phase

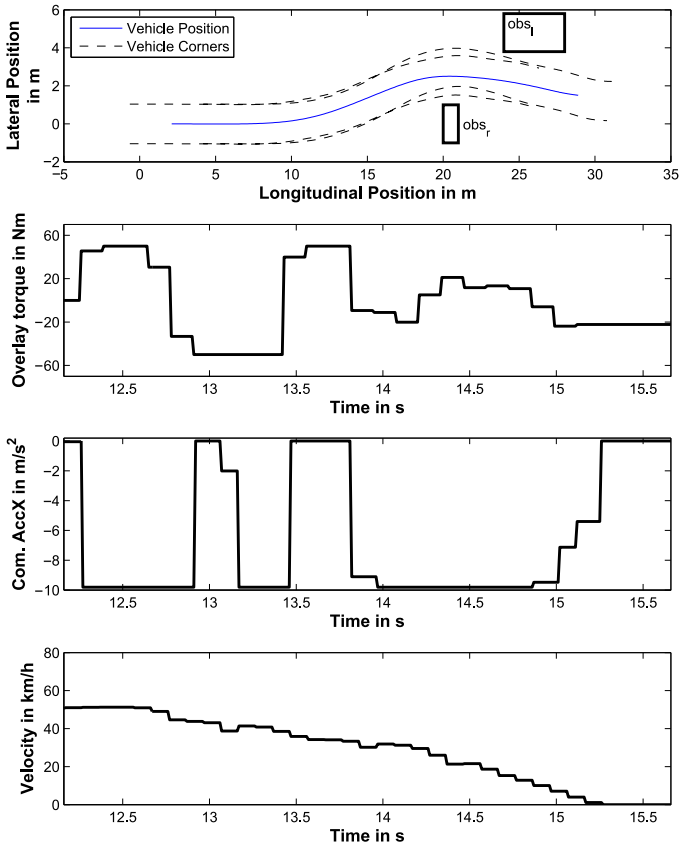



Fig. 7. Collision avoidance in a scenario with multiple obstacles in a real test vehicle.

follows the counter steering phase to steer the vehicle back, whereas in simulation this phase is omitted. The difference can be explained by model deviations between the test vehicle and the single-track model of the planner. In longitudinal dynamics, the deceleration signal oscillates in the first half which may be explained by the rough approximation of the longitudinal dynamics in the planner. Further, Table II shows the metric values of the approximate planning method in real vehicle. The results show that the computation time of the approximate planning method in both simulation and real vehicle are similar, while slight deviations in the velocity reduction and the final yaw angle can be observed. As explained above this can be explained by approximations in the longitudinal model.

## VII. CONCLUSION

This work investigated maneuver decision and trajectory planning for automated driving. Depending on the choice of the vehicle dynamics model, the integrity constraints and the objective function, it has been concluded that the problem may be solved analytically, numerically with a fixed time solver to guarantee optimality, or with nonlinear optimization methods that hardly provide any guarantees. While requiring a good trajectory initialization, path-velocity decomposition has been shown to allow for linear constraints without variation in the velocity profile. A homotopy concept allowing for systematic partitioning of the trajectory space has been introduced. Each

homotopy class corresponds to a specific driving maneuver. Fast solvers are then applied to each homotopy class and the best trajectory among all classes is selected. Particularly for classes that include a critical driving maneuver, nonlinearities need to be considered. Therefore, an objective function for simultaneous longitudinal and lateral trajectories based on a sophisticated vehicle dynamics model and considering integrity and drivability constraints have been formulated firstly. This results in a non-quadratic objective function with nonlinear constraints and coupling effects between lateral and longitudinal dynamics as well as in the constraints. Subsequently, the nonlinear problem has been approximated by a quadratic problem with fixed and linear constraints such that effective optimization algorithms can be used and computation time can be reduced. The proposed methods were able to find the global optimum trajectory even in traffic scenes with multiple vehicles. Experiments with real vehicles demonstrated that the method even allows planning of critical maneuvers like in last moment collision avoidance. Furthermore, the experiments proved the real-time and real-world applicability of the proposed method.

Future work is directed towards real-time application of the proposed method to urban driving scenarios.

## REFERENCES


[1] L. E. Dubins, "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *Amer. J. Math.*, vol. 79, pp. 497–516, 1957.

[2] N. Moshchuk, S. K. Chen, C. Zagorski, and A. Chatterjee, "Optimal braking and steering control for active safety," in *Proc. Int. IEEE Conf. Intell. Transp. Syst.*, 2012, pp. 1741–1746.

[3] R. Hayashi, J. Isogai, P. Raksincharoensak, and M. Nagai, "Autonomous collision avoidance system by combined control of steering and braking using geometrically optimised vehicular trajectory," *Vehicle Syst. Dyn.*, vol. 50, pp. 151–168, 2012.

[4] C. Schmidt, "Fahrstrategien zur Unfallvermeidung im Straßenverkehr für Einzel- und Mehrobjektszenarien," Ph.D. dissertation, Dept. Meas. Control, Karlsruhe Inst. Technol., Karlsruhe, Germany, 2013.

[5] U. Stählin, "Eingriffsentscheidung für ein Fahrerassistenzsystem zur Unfallvermeidung," Ph.D. dissertation, Inst. Automat. Control Mechatron., Tech. Univ. Darmstadt, Germany, 2008.

[6] M. Schorn, "Quer- und Längsregelung eines Personenkraftwagens für ein Fahrerassistenzsystem zur Unfallvermeidung," Ph.D. dissertation, Inst. Automat. Control Mechatron., Tech. Univ. Darmstadt, Germany, 2007.

[7] B. Krogh and C. Thorpe, "Integrated path planning and dynamic steering control for autonomous vehicles," in *Proc. IEEE Int. Conf. Robot. Automat.*, San Francisco, CA, USA, 1986, pp. 1664–1669.

[8] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Int. J. Robot. Res.*, vol. 5, pp. 90–98, 1986.

[9] Y. Rasekhipour, A. Khajepour, S.-K. Chen, and B. Litkouhi, "A potential field-based model predictive path-planning controller for autonomous road vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 5, pp. 1255–1267, May 2017.

[10] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Int. J. Robot. Res.*, vol. 30, pp. 846–894, 2011.

[11] J. Ziegler, P. Bender, T. Dang, and C. Stiller, "Trajectory planning for Bertha—A local, continuous method," in *Proc. IEEE Intell. Vehicles Symp.*, Ypsilanti, MI, USA, 2014, pp. 450–457.

[12] M. Werling and D. Liccardo, "Automatic collision avoidance using model-predictive online optimization," in *Proc. IEEE Conf. Decis. Control*, Maui, HI, USA, 2012, pp. 6309–6314.

[13] B. Gutjahr, L. Gröll, and M. Werling, "Lateral vehicle trajectory optimization using constrained linear time-varying MPC," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 6, pp. 1586–1595, Jun. 2017.

[14] T. Schouwenaars, "Safe trajectory planning of autonomous vehicles," Ph.D. dissertation, Dept. Aeronaut. Astronaut., Massachusetts Inst. Technol., Cambridge, MA, USA, 2005.

[15] M. Werling, B. Gutjahr, S. Galler, and L. Gröll, "Riccati-Trajektorienoptimierung für den aktiven Fußgängerschutz," *at-Automatisierungstechnik*, vol. 63, pp. 202–210, 2015.

[16] J. Ziegler *et al.*, "Making bertha drive—an autonomous journey on a historic route," *IEEE Intell. Transp. Syst. Mag.*, vol. 6, no. 2, pp. 8–20, Summer 2014.

[17] E. Bauer *et al.*, "PRORETA 3: An integrated approach to collision avoidance and vehicle automation," *at-Automatisierungstechnik*, vol. 60, pp. 755–765, 2012.

[18] S. M. Erlien, "Shared vehicle control using safe driving envelopes for obstacle avoidance and stability," Ph.D. dissertation, Dept. Mech. Eng., , Stanford Univ., Stanford, CA, USA, 2015.

[19] J.-P. Laumond, "Feasible trajectories for mobile robots with kinematic and environment constraints," in *Proc. Int. Conf. Intell. Auton. Syst.*, 1986, pp. 346–354.

[20] S. Petti and T. Fraichard, "Safe motion planning in dynamic environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2005, pp. 2210–2215.

[21] X. Qian, F. Altché, P. Bender, C. Stiller, and A. De La Fortelle, "Optimal trajectory planning for autonomous driving integrating logical constraints: An MIQP perspective," in *Proc. IEEE Conf. Intell. Transp. Syst.*, Rio de Janeiro, Brazil, 2016, pp. 205–210.

[22] S. Bhattacharya, M. Likhachev, and V. Kumar, "Search-based path planning with homotopy class constraints," in *Proc. 24th AAAI Conf. Artif. Intell.*, 2010, pp. 1230–1237.

[23] S. Bhattacharya, M. Likhachev, and V. Kumar, "Identification and representation of homotopy classes of trajectories for search-based path planning in 3D," in *Proc. Robot., Sci. Syst.*, 2011, pp. 9–16.

[24] S. Kim, K. Sreenath, S. Bhattacharya, and V. Kumar, "Trajectory planning for systems with homotopy class constraints," in *Proc. 13th Int. Symp. Adv. Robot Kinematics*, 2012, pp. 83–90.

[25] S. Shalev-Shwartz, S. Shammah, and A. Shashua, "On a formal model of safe and scalable self-driving cars," 2017, arXiv preprint arXiv:1708.06374.

[26] Y. Hattori, E. Ono, and S. Hosoe, "Optimum vehicle trajectory control for obstacle avoidance problem," *IEEE/ASME Trans. Mechatronics*, vol. 11, no. 5, pp. 507–512, Oct. 2006.

[27] B. Mashadi and M. Majidi, "Global optimal path planning of an autonomous vehicle for overtaking a moving obstacle," *Latin Amer. J. Solids Struct.*, vol. 11, pp. 2555–2572, 2014.

[28] M. Elbanhawi and M. Simic, "Sampling-based robot motion planning: A review," *IEEE Access*, vol. 2, pp. 56–77, 2014.

[29] Y. Kuwata, G. A. Fiore, J. Teo, E. Frazzoli, and J. P. How, "Motion planning for urban driving using RRT," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nice, France, 2008, pp. 1681–1686.

[30] J. Farinella, C. Lay, and S. Bhandari, "UAV collision avoidance using a predictive rapidly-exploring random tree," in *Proc. AIAA SciTech Forum*, San Diego, CA, USA, 2016, pp. 4–8.

[31] J. P. Timings and D. J. Cole, "Minimum manoeuvre time of a nonlinear vehicle at constant forward speed using convex optimisation," in *Proc. Int. Symp. Adv. Vehicle Control*, Loughborough, U.K., 2010.

[32] M. Pivtoraiko and A. Kelly, "Efficient constrained path planning via search in state lattices," in *Proc. Int. Symp. Artif. Intell., Robot., Automat. Space*, Munich, Germany, 2005, pp. 1–7.

[33] M. Pivtoraiko and A. Kelly, "Differentially constrained motion replanning using state lattices with graduated fidelity," in *Proc. Int. Conf. Intell. Robots Syst.*, Sydney, NSW, Australia, 2008, pp. 2611–2616.

[34] X. Sun, W. Yeoh, and S. Koenig, "Generalized fringe-retrieving A*: Faster moving target search on state lattices," in *Proc. Int. Conf. Auton. Agents Multiagent Syst.*, Toronto, ON, Canada, 2010, pp. 1081–1088.

[35] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," Iowa State Univ., Ames, IA, USA, Tech. Rep. TR 98-11, 1998.

[36] S. M. LaValle and J. J. Kuffner Jr., "Rapidly-exploring random trees: Progress and prospects," in *Proc. Workshop Algorithmic Found. Robot.*, 2000, pp. 293–308.

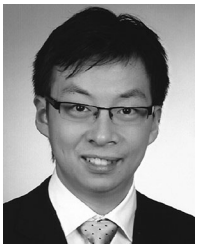
[37] A. Bry and N. Roy, "Rapidly-exploring random belief trees for motion planning under uncertainty," in *Proc. IEEE Int. Conf. Robot. Automat.*, Shanghai, China, 2011, pp. 723–730.

[38] M. Mitschke and H. Wallentowitz, *Dynamik der Kraftfahrzeuge*. Berlin, Germany: Springer, 2004.

[39] A. Takahashi, T. Hongo, Y. Ninomiya, and G. Sugimoto, "Local path planning and motion control for AGV in positioning," in *Proc. IEEE/RSJ Int. Workshop Intell. Robots Syst.*, Tsukuba, Japan, 1989, pp. 392–397.

[40] P. Bender, Ö. Ş. Taş, J. Ziegler, and C. Stiller, "The combinatorial aspect of motion planning: Maneuver variants in structured environments," in *Proc. IEEE Intell. Vehicles Symp.*, Seoul, South Korea, 2015, pp. 1386–1392.

[41] S. W. C. Choi, H. I. Choi, and H. P. Moon, "Mathematical theory of medial axis transform," *Pac. J. Math.*, vol. 72, pp. 404–413, 1997.
[42] C. Pudney, "Distance-ordered homotopic thinning: A skeletonization algorithm for 3D digital images," *Comput. Vis. Image Understanding*, vol. 181, pp. 57–88, 1998.
[43] H. Pacejka, *Tyre and Vehicle Dynamics* (Automotive Engineering). Waltham, MA, USA: Butterworth-Heinemann, 2006.
[44] K. Kritayakirana, "Autonomous vehicle control at the limits of handling," Ph.D. dissertation, Dept. Mech. Eng., Stanford Univ., Stanford, CA, USA, 2012.
[45] C. Beal and J. C. Gerdes, "Model predictive control for vehicle stabilization at the limits of handling," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 4, pp. 1258–1269, Jul. 2013.
[46] M. Choi and S. Choi, "Model predictive control for vehicle yaw stability with practical concerns," *IEEE Trans. Veh. Technol.*, vol. 63, no. 8, pp. 3539–3548, Oct. 2014.



**Boliang Yi** received the M.Sc. degree in mechatronics engineering from the Technical University of Darmstadt, Darmstadt, Germany, in 2013, and the Ph.D. degree in advanced technology from the Opel Automobile GmbH, Rüsselsheim, Germany, in 2017, under the supervision of Prof. C. Stiller from the Institute for Measurement and Control Systems, Karlsruhe Institute of Technology, Karlsruhe, Germany.

**Philipp Bender** received the Diploma degree from the Karlsruhe Institute of Technology, Karlsruhe, Germany, in 2011. He was a Research Assistant with FZI working in the field of maps and trajectory planning for autonomous vehicles. In 2013, he was part of the team working on the Bertha Benz Project in cooperation with Daimler AG.

**Frank Bonarens** received the Diploma degree in electrical engineering from the Technical University Darmstadt, Darmstadt, Germany, in 1990. He gathered profound automotive expertise in vehicle development, passive safety, interior validation, chassis electronics, powertrain interfaces, and advanced engineering with the Opel Automobile GmbH. He was the Opels Project Leader at the German funded project UR:BAN and is working in the field of automated driving.

**Christoph Stiller** received the Diploma degree in electrical engineering and the Ph.D. degree from the RWTH Aachen University, Aachen, Germany, in 1988 and 1994, respectively. He held a Postdoctoral position with INRS, Montreal, QC, Canada. In 1995, he joined the Research of Robert Bosch GmbH in Germany. Since 2001, he has been a full Professor with the Karlsruhe Institute of Technology, Karlsruhe, Germany. He has served the IEEE ITS Society in numerous positions, including as an Editor-in-Chief for the *ITS Magazine*, and the Vice-President and the Society President in 2012 and 2013, respectively.