Safe Nonlinear Trajectory Generation for Parallel Autonomy with a Dynamic Vehicle Model

Wilko Schwarting[1], Javier Alonso-Mora[1,2], Liam Paull[1,3], Sertac Karaman[4], Daniela Rus[1]

***Abstract*—High-end vehicles are already equipped with safety systems, such as assistive braking and automatic lane following, enhancing vehicle safety. Yet, these current solutions can only help in low-complexity driving situations. In this work, we introduce a *Parallel Autonomy*, or shared control, framework that computes safe trajectories for an automated vehicle, based on human inputs. We minimize the deviation from the human inputs while ensuring safety via a set of collision avoidance constraints. Our method achieves safe motion even in complex driving scenarios, such as those commonly encountered in an urban setting. We introduce a receding horizon planner formulated as Nonlinear Model Predictive Control (NMPC), which includes analytic descriptions of road boundaries and the configuration and future uncertainties of other road participants. The NMPC operates over *both* steering and acceleration simultaneously. We introduce a nonslip model suitable for handling complex environments with dynamic obstacles, and a nonlinear combined slip vehicle model including normal load transfer capable of handling static environments. We validate the proposed approach in two complex driving scenarios. First, in an urban environment that includes a left-turn across traffic and passing on a busy street. And second, under snow conditions on a race track with sharp turns and under complex dynamic constraints. We evaluate the performance of the method with various human driving styles. We consequently observe that the method successfully avoids collisions and generates motions with minimal intervention for Parallel Autonomy. We note that the method can also be applied to generate safe motion for fully autonomous vehicles.**

***Index Terms*—Advanced Driver Assistance Systems (ADAS), parallel autonomy, motion planning, collision avoidance, trajectory generation, shared control, intelligent vehicles**

I. Introduction

The article is a revised and extended version of the paper entitled "Parallel Autonomy in Automated Vehicles: Safe Motion Generation with Minimal Intervention" IEEE ICRA 2017, Singapore, June 2017, [1]. The main extensions are a combined-slip dynamical vehicle model including load transfer, the motivation of computationally tractable probabilistic collision constraints for general planning in dynamic and uncertain environments, the generalization of the minimal intervention principle to any human control inputs, a thorough technical discussion including caveats, and in-depth explanations of the NMPC, minimal intervention and the combination thereof, derivations and additional results including a general measure of intervention. The Associate Editor for this paper was XXXXX

[1]Computer Science and Artificial Intelligence Laboratory (CSAIL), MIT, Cambridge, MA, USA {wilkos,rus} @csail.mit.edu

[2]Cognitive Robotics, Delft University of Technology, Delft, Netherlands J.AlonsoMora@tudelft.nl

[3]Département d'informatique et de recherche opérationnelle, Université de Montréal, Montréal, Canada paulll@iro.umontreal.ca

[4]Laboratory of Information and Decision Systems (LIDS), MIT, Cambridge, MA, USA sertac@mit.edu

Toyota Research Institute ("TRI") provided funds to assist the authors with their research but this article solely reflects the opinions and conclusions of its authors and not TRI or any other Toyota entity.

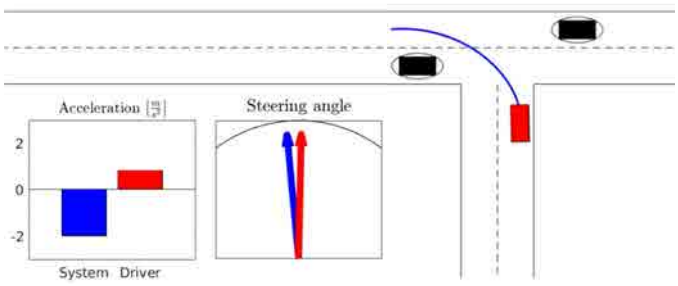


Fig. 1. Parallel Autonomy in complex driving scenarios: Human driver (red) tries to accelerate into an intersection, as shown by the red bar in the lower left inset. However, given the future uncertainty of the other vehicles' positions the Parallel Autonomy system prevents the vehicle from continuing and potentially inhibits a collision.

GLOBALLY, over 3,000 human lives are lost *every day* [2] in vehicle-related accidents and over one hundred thousand are injured or disabled on average. Worse still is that this number is continuing to increase [3]. In the United States, 11% of accidents are caused by driver distraction (such as cell phone use), 31% involve an impaired driver due to alcohol consumption, 28% involved speeding, and an additional 2.6% were due to fatigue [4]. This troubling trend has resulted in the continued development of advanced safety systems by commercial car manufacturers.

For example, systems exist to automatically brake in the case of unexpected obstacles [5], maintain a car in a lane at a given speed, and alert users of pedestrians, signage, and other vehicles on the roadway [6]. However, the scenarios that these systems are able to deal with are relatively simple compared to the diverse and complicated situations that we find ourselves in as human drivers routinely. Human drivers are capable of handling nearly all driving tasks well enough most of the time, but are overwhelmed in key moments when quick and precise actions are needed in fast and complex traffic scenarios. A deer crossing the road, a preceding crash on the highway, a missed blind spot, or driver tiredness are only some of the ubiquitous challenges human drivers face in everyday traffic. To handle these situations, an advanced autonomy system must leave the driver in full control while ensuring safety when required.

In this work we propose a framework for advanced safety in complex scenarios that we refer to as Parallel Autonomy. In this framework we minimize the deviation from the human input while ensuring safety. The design of the system has two main objectives: (a) minimal intervention - we only apply autonomous control when necessary, and (b) guaranteed safety

- the collision free state of the vehicle is explicitly enforced through constraints in the optimization.

There are three types of collaborative autonomy: (1) series autonomy, in which the human driver orders the vehicle to execute a function, similar to most self-driving approaches to date; (2) interleaved autonomy, in which the human driver and the autonomous system intermittently take turns in operation of the vehicle; and (3) Parallel Autonomy, in which the autonomous system functions as a guardian angel in the background to ensure safety while the human driver is operating the vehicle. Whether drivers are distracted by smartphones, searching in their glove box, operating the navigation system, or are simply overwhelmed by the difficulty of driving in challenging scenarios, the Parallel Autonomy principles offer additional safety due to redundancy.

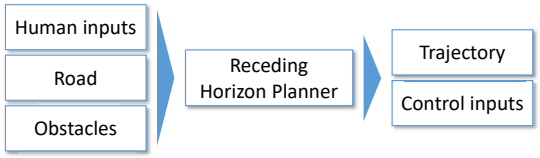


Fig. 2. A receding horizon planner computes control inputs based on human inputs and an environment consisting of the road, and static and dynamic obstacles.

We provide a formulation and algorithmic solution to Parallel Autonomy based on a Nonlinear Model Predictive Control (NMPC) policy, under the assumption of known current configuration of the ego vehicle and the road boundaries, cf. Fig.2. Uncertain current and future configurations of other vehicles are described in the form of a posterior distribution and we parametrize them by their mean, the expected configuration, and covariances. We assume this information to be available from an inference framework. Specifically,

- we incorporate the time-varying uncertainty related to the dynamic obstacle predictions explicitly in the optimization,
- the vehicle can follow the road by contour tracking with additional constraints for the road boundaries,
- and we simultaneously optimize over steering and acceleration while maintaining the ability to plan online over long time horizons ($\sim 9s$),

The basic operation of the controller is shown in Fig. 1, where the driver attempts to cut in front of oncoming traffic to make a left turn, however the Parallel Autonomy system prevents this action to avoid a collision with the incoming vehicles.

This paper contributes the following:

- A formulation of Parallel Autonomy as a shared control approach between humans and intelligent vehicles, which adheres to the minimal intervention principle and is able to handle complex driving scenarios
- The development of a real-time NMPC suitable for trajectory generation in intelligent and autonomous vehicles, which relies on a state of the art solver[1]
- Simulation of complex traffic scenarios with real human inputs of different driving styles, such as a left-turn across traffic and driving on a snowy race track

We will employ two motion models of different complexity:

- A dynamical nonlinear combined slip vehicle model including load transfer for static environments
- A kinematic model for dynamic and complex environments

In Sec. II we summarize the related work in the field whereas in Sec. III we present the Parallel Autonomy control approach. In Sec. IV we provide a concrete instantiation of the framework and present the NMPC approach to solve it. Finally, we show detailed simulation results and conclusions in Sec. V and Sec. VI respectively.

[1]We employ FORCES Pro by Embotech to autogenerate a fast NMPC solver for our problem.

II. Related Works

In this section we provide an overview of the related work in the areas of general safety, shared control for autonomous vehicles, and Model Predictive Control (MPC).

A. Safety of Autonomous Vehicles

In theory, safety can be guaranteed for deterministic systems by computing the set of the states for which the vehicle will *inevitably* have a collision and then ensuring that the vehicle never enters that set. The set is referred to by different terms in the literature, such as the capture set [7], [8], [9], [10], the inevitable collision states (ICS) [11], [12], [13], the region of inevitable collision (RIC) [14], and the target set [15]. However, without some assumptions or limiting the applicability to relatively simplistic scenarios, this set is difficult to compute analytically. [16] apply a control barrier function to guarantee never entering the infeasible set upon moving into an avoidable set constructed from a polar algorithm in slow speeds to avoid pedestrians. These ICS-inspired methods tend to only intervene when the system is at the boundary of the capture set, which can cause undesirable behavior, and toggle between either the autonomous system input or the human input. We follow the idea of [7], [12], [13] and define a set of probabilistic constraints for collision avoidance, which produce a safe behavior. Yet, our method produces smooth inputs for the vehicle, since it is always active, as a Parallel Autonomy safety system.

B. Shared Control of Intelligent Vehicles

The most intuitive way of merging the human input with the output of a safety system is by linear combination of the two, as shown by [17] and [18], who proposed threat measures based on the dynamic limitations of the vehicle. The human input was overwritten by the trajectory based on the severity of the threat. While they reasoned about the human drivers intended homotopy class [17], their framework did not take the human input directly into account. In a similar line, [19] computed safety margins from sampled trajectories clustered into homotopy classes, but do not share the control with the human driver.

In contrast, in this work we directly incorporate the human inputs into an optimization framework in a minimally invasive manner and also add a soft *nudging* behavior to guide the driver. Our approach minimizes the deviation of the autonomous system's plan from the driver's intent - current steering and acceleration inputs - and provides small feedback to the driver shortly before situations become critical. This is important to prevent startling the driver and to decrease the likelihood of unnecessary high intervention later, thanks to slight intervention early on.

Similar to our approach, several constrained optimization approaches for shared control have been formulated. [20] directly minimized the difference from the human predicted control input necessary to achieve safe trajectories, and [21] minimized the difference in steering wheel angle. [22] minimize the deviation from desired front wheel lateral force with an additional discount factor with increasing time. We apply a discount factor in a similar manner. [23] minimized the deviation from human inputs, in this case orientation and speed, via a convex constrained optimization to generate safe motion of a wheelchair. In contrast, we present a more general approach where we conduct the minimization jointly in *both* steering and acceleration inputs, can blend in additional trajectory-specific costs to provide a nudging behavior, model the dynamical model of the vehicle, and strictly enforce safety constraints.

C. Receding Horizon Control for Shared Control

We formulate the constrained optimization as a receding horizon control problem, typically referred to as Model Predictive Control (MPC). Related to our work, [21] employed a hierarchical MPC approach for avoidance of static obstacles with motion primitives and path tracking, which switches control to and from the driver as a function of driver attentiveness. Instead of planning paths first and computing velocity profiles separately, we directly plan full trajectories that also avoid dynamic obstacles. Similar approaches include: a constrained pathless MPC that blends human and controller steering commands only, proposed by [17], a shared control MPC for static unstructured robot environments avoiding circular obstacles [24], and robust NMPC [21] to avoid static obstacles while tracking the roads center line over a very short horizon of less than $1.5s$. Alternatively, [22] defined vehicle-stability and environmental envelopes to supply safe steering commands at constant speed in a discretized environment.

In contrast, our approach does not require linearization - we solve a Nonlinear MPC (NMPC) problem directly - and can handle complex road scenarios with dynamic maneuvers and dynamic obstacles, with steering *and* acceleration control over long horizons ($\sim 9s$).

The methods developed by [25] and [22] construct corridors consisting of multiple convex regions to describe the area that the vehicle will drive through. Based on the corridors and constant velocity, they apply constraints on the lateral position of the vehicle to avoid colliding with obstacles yielding a convex optimization problem with global optimality. Operating over both steering and acceleration in a non-convex environment and solving our NMPC formulation inherits the general limitations of non-convex optimization, such as uncertain convergence and runtime, and lack of guarantee of optimality. Nonetheless, braking is clearly an essential function in vehicle safety.

We provide all costs and constraints to the solver in closed form without pre-linearization, and we benefit from recent advances in efficient Interior-Point solvers [26] to directly solve the NMPC. To guide the planner along the road, we build upon Model Predictive Contouring Control (MPCC) [27], [28], [29], which approximates path progress inside a corridor, the road in our application. By tracking the center of the lane and remaining within the limits of the road our planner can be employed for both Parallel Autonomy, where we minimize the deviation from human input, and for fully autonomous vehicles.

III. Problem formulation

The Parallel Autonomy problem is based on two overarching principles.

- *Minimal intervention* with respect to the human driver. That is, the control inputs to the vehicle should be as close as possible to those of the human driver.
- *Safety*. The probability of collision with respect to the environment and other traffic participants is below a given threshold.

A. Definitions

We use the discrete time shorthand $k \triangleq t_k$, where $t_k = t_0 + \sum_{i=1}^{k} \Delta t_i$, with t_0 the current time and Δt_i the i-th timestep of the planner. Vectors are bold.

1) Ego vehicle: We refer to the car driven by the human as ego-vehicle. At time k, we denote the configuration of the ego-vehicle, typically position $\mathbf{p}_k = [x_k, y_k]$, heading ϕ_k, longitudinal and lateral velocity $v_{x,k}$, $v_{y,k}$, yaw rate $\dot{\phi}_k$ and steering angle δ_k, by the state $\mathbf{z}_k = [\mathbf{p}_k, \phi_k, \delta_k, v_{x,k}, v_{y,k}] \in \mathcal{Z}$. Its control input, typically steering velocity $\dot{\delta}_k$ and longitudinal acceleration $\dot{v}_{x,k}$, is labeled $\mathbf{u}_k = [\dot{\delta}^u_k, \dot{v}^u_{x,k}] \in \mathcal{U}$.

The evolution of the state of a vehicle is then represented by a general discrete dynamical system

$$\mathbf{z}_{k+1} = f(\mathbf{z}_k, \mathbf{u}_k), \tag{1}$$

described in Sec. III-C2. The arguably largest source of uncertainty in the system operating over long time horizons lies in the prediction of motions of other vehicles, which dominates all other sources of uncertainty. The comparably low uncertainty in the dynamic motions of the ego-vehicle thus motivates the choice for a deterministic motion model. The area occupied by the ego-vehicle at state $\mathbf{z}_k$ is denoted by $\mathcal{B}(\mathbf{z}_k) \subset \mathbb{R}^2$. In particular, we model it as a union of circles as shown in Sec. IV-E, Fig. 5.

2) Other traffic participants: Other traffic participants, such as vehicles, pedestrians and bikes, are indexed by $i = \{1, \dots, n\}$. Their configuration and control input are described as $\mathbf{z}^i_k \in \mathcal{Z}_i$ and $\mathbf{u}^i_k \in \mathcal{U}_i$. To incorporate uncertainty, we consider that an approximate posterior distribution describing the current and future state of the other vehicles for up to m timesteps is available, e.g. from an inference framework such

as [30]. The distributions are parametrized by their expected mean configuration $[\mathbf{p}^i_{0:m}, \phi^i_{0:m}]$ and covariance $\mathbf{\Sigma}^i_{0:m}$.

At a given state, each traffic participant occupies an area $\mathcal{B}^i(\mathbf{z}^i_k, \mathbf{\Sigma}^i_k, p_\epsilon) \subset \mathbb{R}^2$ with probability larger than p_ϵ. Here p_ϵ is the accepted probability of collision. We model them as ellipses that grow in size with uncertainty, as described in the forthcoming Sec. IV-E to achieve a closed form description of a probabilistic collision constraint and therefore computational tractability.

3) Free space: We consider the workspace $\mathcal{W} = \mathbb{R}^2$ and an obstacle map $\mathcal{O} \subset \mathcal{W}$ containing the static obstacles, such as the limits of the road and areas not classified as road. We define the environment $\mathcal{E}(k)$ as the state of the world (obstacles, traffic participants) at a time instance k.

B. Parallel Autonomy Formulation

We formulate a general discrete time constrained optimization with m timesteps and time horizon $\tau = \sum_{k=1}^{m} \Delta t_k$. We use the following notation for a set of states $\mathbf{z}_{0:m} = [\mathbf{z}_0, \dots, \mathbf{z}_m] \in \mathcal{Z}^{m+1}$ and for a set of inputs $\mathbf{u}_{0:m-1} = [\mathbf{u}_0, \dots, \mathbf{u}_{m-1}] \in \mathcal{U}^m$.

The objective is to compute the optimal inputs $\mathbf{u}^*_{0:m-1}$ for the ego-vehicle which minimize a cost function $\hat{J}_h(\mathbf{u}_{0:m-1}, \mathbf{u}^h_0) + \hat{J}_t(\mathbf{z}_{0:m}, \mathbf{u}_{0:m-1})$, where

- $\hat{J}_h(\mathbf{u}_{0:m-1}, \mathbf{u}^h_0)$ is a cost term that minimizes the deviation from the currently observable human input $\mathbf{u}^h_0$.
- $\hat{J}_t(\mathbf{z}_{0:m}, \mathbf{u}_{0:m-1})$ is a cost term that only depends on intrinsic properties of the planned trajectory. It can include various optimization objectives such as energy minimization, comfort, or following a lane.

The optimization is subject to a set of constraints, which represent:

- the transition model of the ego-vehicle, $\mathbf{z}_{k+1} = f(\mathbf{z}_k, \mathbf{u}_k)$,
- no collisions with static obstacles, $\mathcal{B}(\mathbf{z}_k) \cap \mathcal{O} = \emptyset$, and
- no collisions with other traffic participants up to probability p_ϵ described by $\mathcal{B}(\mathbf{z}_k) \cap \bigcup_{i \in \{1,\dots,n\}} \mathcal{B}^i(\mathbf{z}^i_k, \mathbf{\Sigma}^i_k, p_\epsilon) = \emptyset$.

Given the estimated trajectories $(\mathbf{z}^i_{0:m}, \mathbf{\Sigma}^i_{0:m})$ for all traffic participants $i = 1, \dots, n$ and the initial state $\mathbf{z}_0$ of the ego-vehicle, the optimal trajectory for the ego-vehicle is then given by the following receding-horizon optimization,

$$\begin{aligned} \mathbf{u}^*_{0:m-1} = \underset{\mathbf{u}_{0:m-1}}{\arg\min}\ & \hat{J}_h(\mathbf{u}_{0:m-1}, \mathbf{u}^h_0) + \hat{J}_t(\mathbf{z}_{0:m}, \mathbf{u}_{0:m-1}) \\ \text{s.t. } & \mathbf{z}_{k+1} = f(\mathbf{z}_k, \mathbf{u}_k) \\ & \mathcal{B}(\mathbf{z}_k) \cap \mathcal{O} = \emptyset \\ & \mathcal{B}(\mathbf{z}_k) \cap \bigcup_{i \in \{1,\dots,n\}} \mathcal{B}^i(\mathbf{z}^i_k, \mathbf{\Sigma}^i_k, p_\epsilon) = \emptyset, \\ & \qquad\qquad \forall k \in \{0, \dots, m\}. \end{aligned} \tag{2}$$

We describe the method in detail in Sec. IV.

C. Motion Models

Previous approaches utilized constant longitudinal speed and small angle assumptions in selected static obstacle avoidance scenarios along straight roads [17], [18], [22]. In contrast, we consider the impact of joint speed and steering control for higher safety in dynamic, more general and more complex traffic environments over longer time horizons.

TABLE I
MAIN SYMBOLS.

$\mathbf{z} \in \mathcal{Z}, \mathbf{u} \in \mathcal{U}$	Vehicle state and control input
J_h, J_t	Minimal intervention, and trajectory cost
$\mathcal{B}^i(\mathbf{z}^i_k, \Sigma^i_k, p_\epsilon)$	Probabilistic footprint of dynamic obstacles
$\mathcal{B}(\mathbf{z}_k)$	Ego-vehicle footprint
$\mathcal{O}, \mathcal{E}, \mathcal{W}$	Static obstacles, environment, workspace

We first introduce a kinematic motion model with constraints to ensure limited slip for complex and dynamic environments, and subsequently a more complex combined slip dynamic model including load transfer. Our method applies to both models.

1) Kinematic Model: The kinematic model is a simplified car model with a fixed rear wheel and a steerable front wheel with state $\mathbf{z}$ and controls $\mathbf{u} = [\dot{\delta}^u, \dot{v}^u_x]$ with lateral velocity $v_y = 0$. The rear-wheel driven vehicle with inter-axle distance L and continuous kinematic model

$$\underbrace{\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \\ \dot{\delta} \\ \dot{v}_x \end{bmatrix}}_{\dot{\mathbf{z}}} = \begin{bmatrix} v_x \cos(\phi) \\ v_x \sin(\phi) \\ \frac{v_x}{L} \tan(\delta) \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} \dot{\delta}^u \\ \dot{v}^u_x \end{bmatrix}}_{\mathbf{u}}, \tag{3}$$

is described by a discrete time model by integration $\mathbf{z}_{k+1} = \mathbf{z}_k + \int_k^{k+\Delta t_k} \dot{\mathbf{z}}\, \mathrm{d}t = f(\mathbf{z}_k, \mathbf{u}_k)$. A fourth order Runge-Kutta scheme ensures integration between timesteps k to sufficient accuracy.

We limit steering angle, $|\delta| \leq \delta_{\max}$, steering speed, $|\dot{\delta}| \leq \dot{\delta}_{\max}$, longitudinal speed, $v_x \leq v_{x,\max}$, as well as extreme breaking and accelerations $\dot{v}_{x,\min} \leq \dot{v}_x \leq \dot{v}_{x,\max}$ to reasonable values conforming to vehicle performance and the rules of the road, e.g. speed limits.

We account for, and prohibit, unsafe driving modes such as high cornering speeds by limiting the product of longitudinal velocity and yaw-rate

$$|v_x \dot{\phi}| \leq (v_x \dot{\phi})_{\max}, \tag{4}$$

which essentially poses a velocity dependent constraint on the vehicle's maximum allowed curvature. This model works well for low speeds or less aggressive driving behaviors.

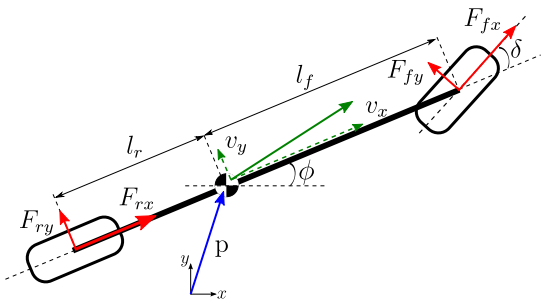


Fig. 3. Dynamical half-car model.

2) Combined Slip Model with Load Transfer: Combined breaking and steering is one of the most essential aspects of vehicle safety and motivates our choice for a combined slip model [31]. Specifically, we allow load transfer between the front and rear tires - a technique often used by rally racing drivers to control the yaw dynamics [32]. For simplicity, we neglect suspension and wheel dynamics, and assume that only the front wheel is steerable. The motion model is a dynamical half-car model with mass m, and yaw moment of inertia I_z. The distances of front and rear wheel from the center of gravity (**cog**) are l_f and l_r respectively. h is the height of the **cog**. The position of the **cog** in inertial frame is given by $\mathbf{p} = [x, y]$, the heading by ϕ, and v_x, v_y denote the longitudinal and lateral velocities in the car-body fixed axis system, and $F_{\alpha\beta}$, $\alpha \in \{f, r\}$, $\beta \in \{x, y\}$, the tire force components, cf. Fig. 3. The equations of motion of the dynamical half-car model are then given by

$$\dot{\mathbf{p}} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} v_x \\ v_y \end{bmatrix}, \tag{5}$$

$$\begin{bmatrix} m\dot{v}_x \\ m\dot{v}_y \\ I_z\ddot{\phi} \end{bmatrix} = \begin{bmatrix} F_{fx}\cos\delta - F_{fy}\sin\delta + F_{rx} + mv_y\dot{\phi} \\ F_{fx}\sin\delta + F_{fy}\cos\delta + F_{ry} - mv_x\dot{\phi} \\ l_f\left(F_{fx}\sin\delta + F_{fy}\cos\delta\right) - l_rF_{ry} \end{bmatrix}; \tag{6}$$

and are described by a discrete time model by integration in the same way as for the kinematic model.

We denote the state as $\mathbf{z} = [\mathbf{p}, \phi, \delta, v_x, v_y]$, control the vehicle by steering velocity $\dot{\delta}$, and longitudinal slip control s, such that $\mathbf{u} = [\dot{\delta}^u, s^u]$. s^u defines the longitudinal slips as $s_{fx} = \gamma(s^u)s^u$, $s_{rx} = (1 - \gamma(s^u))\, s^u$, with $\gamma(s^u) \in [0, 1]$ specifying the longitudinal slip distribution among front and rear tire. This enables us to model both front and rear wheel driven vehicles as well as arbitrary all-wheel drive configurations.

The lateral slips for front and rear tire are

$$s_{fy} = \frac{(v_y + l_f\dot{\phi})\cos\delta - v_x\sin\delta}{v_x\cos\delta + (v_y + l_f\dot{\phi})\sin\delta}, \quad s_{ry} = \frac{v_y - l_r\dot{\phi}}{v_x}, \tag{7}$$

and the total tire slips $s_\alpha = \sqrt{s_{\alpha x}^2 + s_{\alpha y}^2}$, $\alpha \in \{f, r\}$. We can compute the tire forces as

$$F_{\alpha\beta} = \mu_{\alpha\beta}F_{\alpha z},\ \alpha \in \{f, r\},\ \beta \in \{x, y\} \tag{8}$$

with normal loads on front and rear tires

$$F_{fz} = \frac{mg(l_r - \mu_{rx}h)}{l_f + l_r + h(\mu_{fx}\cos\delta - \mu_{fy}\sin\delta - \mu_{rx})}, \tag{9}$$

$$F_{rz} = mg - F_{fz}. \tag{10}$$

The corresponding friction coefficients are given by Pacejka's simplified magic formula [33]

$$\mu_{\alpha\beta} = -(s_{\alpha\beta}/s_\alpha)\mu_\alpha,\ \ \alpha \in \{f, r\},\ \ \beta \in \{x, y\} \tag{11}$$

with

$$\mu_\alpha = D_\alpha \sin\left(C_\alpha \arctan(B_\alpha s_\alpha)\right),\ \ \alpha \in \{f, r\}. \tag{12}$$

We limit the total friction forces for both front and rear tires by

$$F_\alpha = \sqrt{F_{\alpha x}^2 + F_{\alpha y}^2} \le \mu_{\alpha,\max}F_{\alpha z},\ \ \alpha \in \{f, r\}, \tag{13}$$

to ensure operation in a safe driving envelope. $\mu_{\alpha,\max}$ is the maximum allowed friction coefficient. The constraint essentially limits the yaw rate dependent on the state $\mathbf{z}$, prohibiting unsafe driving modes suitable even for race-car driving under significant amounts of slip [31].

The constraints on steering speed $|\dot{\delta}| \le \dot{\delta}_{\max}$, steering angle $|\delta| \le \delta_{\max}$, longitudinal speed, $v_x \le v_{x,\max}$, remain the same with an additional constraint on the lateral velocity $|v_y| \le v_{y,\max}$.

IV. Method

In this section we describe the method to solve the general problem of Eq. (2) in a specific setting.

A. Overview

We formulate a NMPC to compute a safe trajectory for the predefined time horizon. The constrained optimization consists of the following costs and constraints.

1) Cost: To maintain generality of the problem formulation while easing the understanding of the specifics of the instantiation, the notation of $\hat{J}_h$, $\hat{J}_t$, cf. Eq. (2), will be slightly altered to J_h, J_t, cf. Eq. (44).

The cost term $J_t(\mathbf{z}_{0:m}, \mathbf{u}_{0:m-1})$ defined in Sec. IV-C consists of terms responsible for giving feedback to the driver in the form of slightly *nudging* the driver back into the correct direction without strong intervention, if diverted too far from the road's center (Sec. IV-B), or if high future control inputs are necessary to maintain safety (Sec. IV-C). This is to avoid unnecessary high future intervention by small intervention early on.

The cost term J_h penalizes the deviation of the system from the current acceleration and steering angle specified by the human driver. This term and its generalization to higher order states is described in Sec. IV-F.

We define the time-dependent weighted combination of J_h and J_t to avoid the necessity of prediction of future human inputs in Sec. IV-G.

2) Constraints: The optimization is subject to a set of constraints: (1) to respect the transition model of the system, including kinematic and dynamic constraints, described in Sec. III-C2, (2) to maintain the vehicle within the limits of the road, indicated in Sec. IV-D and (3) to avoid other traffic participants in the sense of guaranteeing a probability of collision below p_ϵ, as given in Sec. IV-E.

3) Constrained Optimization: The resulting NMPC, which solves Eq. (2), is then described in Sec. IV-H.

B. Lane tracking

In this section we build on the MPCC method of [27], [28], [29] and apply it to our problem setting. The control framework combines path generation and path tracking into one problem. The MPCC essentially plans a progress-optimal path by taking into account the (nonlinear) projection of the vehicles position onto the center line. This is different from tracking controllers in that the controller has more freedom to determine the state trajectories to follow the given path,

for example to schedule the velocity, which in tracking is defined by the reference trajectory. The resulting controller is able to plan and to follow a path inside a specified corridor which is similar to time-optimal paths in race tracks when the horizon is chosen long enough. We are able to apply the MPCC control framework to our problem formulation by including additional cost terms into the optimization such as the minimal intervention cost, cf. Sec. IV-F. We use the center line as a reference path, but employ it merely as a measure of progress by selecting appropriate weights in the optimization.

The MPCC approach is a suitable choice for our Parallel Autonomy formulation since it can (a) ensure safety by staying within the *contour* of the road, (b) can integrate slight *nudging* by penalizing deviation from the center line, and (c) integrate the driver's intention of achieving progress along the road.

Before posing the adapted MPCC problem and integrating it into our approach, we introduce several preliminaries, including the parametrization of the reference path (for which we will use the center line of the driving lane), and the definition of useful error measures due to an approximated path progress.

1) Progress on reference path: The vehicle at position $\mathbf{p}_k = (x_k, y_k)$ at time k tracks a continuously differentiable and bounded two-dimensional geometric reference path $(x^p(\theta), y^p(\theta))$ of path parameter θ with

$$\mathbf{t}(\theta) = \begin{bmatrix} \frac{\partial x^P(\theta)}{\partial \theta} \\ \frac{\partial y^P(\theta)}{\partial \theta} \end{bmatrix}, \ \mathbf{n}(\theta) = \begin{bmatrix} -\frac{\partial y^P(\theta)}{\partial \theta} \\ \frac{\partial x^P(\theta)}{\partial \theta} \end{bmatrix} \tag{14}$$

the tangential and normal vectors. The direction of the path is given by

$$\phi^p(\theta) = \arctan\left(\frac{\partial y^p(\theta)}{\partial x^p(\theta)}\right), \tag{15}$$

and we will refer to $\Delta\phi = \phi - \phi^p(\theta)$ as the deviation of the vehicle's heading to the path. The path is parametrized by the arc-length r, such that $\partial\theta/\partial r = 1$, which allows us to estimate the progress of the vehicle along the reference path with the vehicle's actually driven longitudinal velocity $v_{x,k}$ and distance $r = \int v_x \, \mathrm{d}t$, for small $\Delta\phi$. While parametrization of curves by the arc-length is not trivial, if the distance between knots is small in relation to their curvature, spline parametrization is close to the arc-length. Since the ego-vehicle will follow a given road with sufficiently low deviation from the reference, enforced by the road's boundary, we can assume that

$$\Delta\theta \approx \Delta r = v_x \Delta t \tag{16}$$

holds. This additional assumption yields an approximated progress along the path parameter

$$\theta_{k+1} = \theta_k + v_{x,k}\Delta t_k \tag{17}$$

where $v_{x,k}\Delta t_k$ describes the approximated progress at time step k. Ideally, we want to compute the path parameter $\theta^P(x_k, y_k)$ of the closest point on the reference path to the ego-vehicle's current position $\mathbf{p}_k = (x_k, y_k)$ to evaluate the associated path costs and constraints.

Finding $\theta^P(x_k, y_k)$ analytically from the projection operator involves embedding the optimization

$$\theta^p(x_k, y_k) = \arg\min_{\theta'_k} \left(x_k - x^P(\theta'_k)\right)^2 + \left(y_k - y^P(\theta'_k)\right)^2 \tag{18}$$

which is computationally expensive in the general case and renders the direct projection operator unsuitable for fast optimization. Therefore, we approximate $\theta^P(x_k, y_k)$ by Eq. (17).

2) Longitudinal Error: The approximation of the curvilinear abscissa $\theta^P(x_k, y_k)$ by θ_k introduces two errors (cf. Fig. 4) if the vehicle's actual path deviates from the reference path: a longitudinal (lag) error along the path and a lateral (countouring) error normal to the path. The longitudinal error is

$$\begin{aligned} \tilde{e}^l(\mathbf{z}_k, \theta_k) &= \frac{\mathbf{t}(\theta_k)^T}{||\mathbf{t}(\theta_k)||} \begin{bmatrix} x_k - x^P(\theta_k) \\ y_k - y^P(\theta_k) \end{bmatrix} && (19) \\ &= -\cos\phi^P(\theta_k)\left(x_k - x^P(\theta_k)\right) && (20) \\ &\quad - \sin\phi^P(\theta_k)\left(y_k - y^P(\theta_k)\right) && (21) \end{aligned}$$

projecting the position error of the vehicle with respect to the path's abscissa θ_k along the path's tangent $\mathbf{t}(\theta_k)$, see Fig. 5.

For sufficiently small $\tilde{e}^l(\mathbf{z}_k, \theta_k)$ the approximated path progress is close to the actual curvilinear abscissa (Eq. (16)), and $\theta_k \approx \theta^P(x_k, y_k)$. The longitudinal error $\tilde{e}^l(\mathbf{z}_k, \theta_k)$ needs to be penalized in the MPCC optimization to keep the error of the approximated evolution θ_k along the path sufficiently small, as suggested in [28].

3) Contouring Error: The contouring error

$$\begin{aligned} \tilde{e}^c(\mathbf{z}_k, \theta_k) &= \frac{\mathbf{n}(\theta_k)^T}{||\mathbf{n}(\theta_k)||} \begin{bmatrix} x_k - x^P(\theta_k) \\ y_k - y^P(\theta_k) \end{bmatrix} && (22) \\ &= \sin\phi^P(\theta_k)\left(x_k - x^P(\theta_k)\right) && (23) \\ &\quad - \cos\phi^P(\theta_k)\left(y_k - y^P(\theta_k)\right) && (24) \end{aligned}$$

describes the deviation of the vehicle's actual position from the estimated position projected onto the path's normal. It is thus a good measure of how far the vehicle deviates from a given reference path in lateral direction.

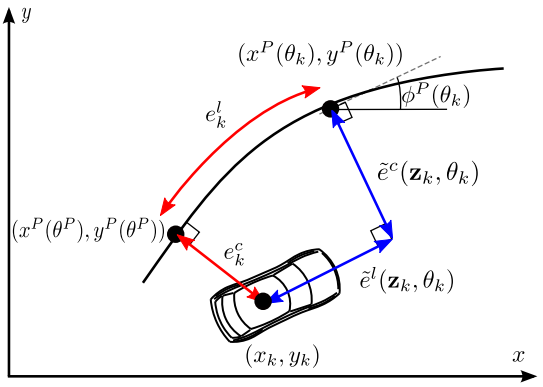


Fig. 4. Approximation of actual path abscissa θ^P by virtual integrator θ_k, and therefore approximation of the true projection $(x^P(\theta^P), y^P(\theta^P))$ by $(x^P(\theta_k), y^P(\theta_k))$. The path projection estimation causes the longitudinal error e_k^l approximated by $\tilde{e}^l(\mathbf{z}_k, \theta_k)$, and contouring error e_k^c approximated by $\tilde{e}^c(\mathbf{z}_k, \theta_k)$. The contouring error $\tilde{e}^c(\mathbf{z}_k, \theta_k)$ is also used for an approximation of the lateral distance of the vehicle to the reference path.

The MPCC cost function

$$J_{\text{MPCC}}(\mathbf{z}_k, \theta_k) = \mathbf{e}_k^T Q \mathbf{e}_k - \rho v_{x,k} \tag{25}$$

with path error vector, formed from approximated longitudinal and contouring error

$$\mathbf{e}_k = \begin{bmatrix} \tilde{e}^l(\mathbf{z}_k, \theta_k) \\ \tilde{e}^c(\mathbf{z}_k, \theta_k) \end{bmatrix}, \tag{26}$$

balances the trade-off between contouring error, longitudinal error, and approximated path progress v_k. Tuning the parameters $Q \in \mathbb{S}^2_{++}$ and $\rho \in \mathbb{R}_+$ yields the direct opportunity to encode the previously discussed nudging if the vehicle diverts too far from the reference path, and the driver's anticipated intention to make progress along the road. Since each stage k in the complete cost function will be scaled with Δt_k, see (44) in Sec. IV-H, $v_{x,k}$ is not multiplied with Δt_k but still represents the approximate progress along the path.

C. Trajectory Cost

The trajectory cost contains the MPCC cost, cf. Eq. (25), and additionally penalizes strong control inputs and yaw rate, a measure of driving comfort

$$J_t(\mathbf{z}_k, \mathbf{u}_k, \theta_k) = J_{\text{MPCC}}(\mathbf{z}_k, \theta_k) + \mathbf{u}_k^T R \mathbf{u}_k + \dot{\phi}_k \alpha \dot{\phi}_k. \tag{27}$$

Weights $R \in \mathbb{S}_{++}$ and $\alpha \in \mathbb{R}_+$ allow for different prioritization. J_{MPCC} readily encodes the penalization of large deviation from the reference path and simultaneously encourages progress along the road which aligns with the driver's goal of progressing towards a destination.

By not only minimizing the intervention J_h but also including the trajectory costs J_t into the total cost function we achieve the following:

1) Continuous increase in intervention: Potentially startling and confusing the driver by abrupt intervention may result in increased risk of degenerated driver performance. Instead of suddenly taking over control in a thrashing manner to enforce safety constraints we achieve a continuous increase in intervention by including the trajectory cost. Therefore, it is possible to provide feedback to the driver by inducing a slight *nudging* behavior, potentially increasing driver alertness and preparation, while continuously increasing the level of intervention the closer the system moves towards meeting safety constraints.

2) Prevent unnecessary high intervention by small early intervention: Early and small intervention resulting in *nudging* the driver into beneficial directions also reduces the risk of large intervention in the future. Additionally to this direct effect some hazardous situations may be completely avoided due to potentially increased awareness and early feedback to the driver. Therefore J_t also aligns with our goal to minimize the occurrence of extreme intervention.

3) Guide the optimizer: Finally, introducing the trajectory cost improves convergence of the optimizer by providing gradients towards optimality.

D. Road Representation and Static Obstacle Constraints

The ego vehicle's reference path is parametrized by C^1-continuous clothoids following the road network through prespecified points. We approximate the clothoids by cubic-splines of closely spaced knots parameterizing the spline by the arc-length to sufficient accuracy. Whereas evaluation of clothoids is computationally expensive, cubic splines provide an analytical parametrization of the reference path, boundaries of the road, and their derivatives needed for solving the nonlinear optimization.

The signed lateral distance $d(\mathbf{z}_k, \theta)$ of the ego vehicle's position $\mathbf{p}_k$ to the reference path is given by the projection along the normal of the reference path at the actual curvilinear abscissa θ^P, again approximated by θ_k such that $d(\mathbf{z}_k, \theta_k) = \tilde{e}^c(\mathbf{z}_k, \theta_k)$.

The free and drivable space of the ego vehicle at the path abscissa θ_k is limited by both the left road boundary $b_l(\theta_k)$ and the right road boundary $b_r(\theta_k)$, cf. Fig. 5, which are parametrized by cubic splines to enable analytic evaluation and derivation. The boundaries may enclose all static obstacles $\mathcal{O}$, such as the limits of the road but also other static obstacles such as construction zones or potentially dangerous objects.

To ensure that the ego vehicle stays clear of all static obstacles the allowed lateral offset to the path is limited by

$$b_l(\theta_k) + w(\Delta\phi_k) \leq d(\mathbf{z}_k, \theta_k) \leq b_r(\theta_k) - w(\Delta\phi_k), \tag{28}$$

where $w(\Delta\phi_k) = w/2 \cos(\Delta\phi_k) + \max(l_f, l_r) \sin(|\Delta\phi_k|)$ is the projection of the vehicle's outline onto the reference path's normal. $w(\Delta\phi_k)$ is larger than half the vehicle's width, since the ego vehicle's relative orientation to the path needs to be accounted for, e.g. in turns. Simply taking the vehicle's length as upper bound turns out to be too conservative for navigating in narrow road segments and would severely limit the host vehicle's capability to utilize the road to its full extent to e.g. avoid another vehicle. We constrain the difference between the ego vehicle's heading ϕ_k and the path's heading $\phi^P(\theta_k)$

$$|\phi_k - \phi^P(\theta_k)| \leq \Delta\phi_{\max} \tag{29}$$

to maintain validity of a less conservative $w(\Delta\phi_k)$ as an upper bound on lateral projection of the ego-vehicle's occupancy $\mathcal{B}(z_k)$.

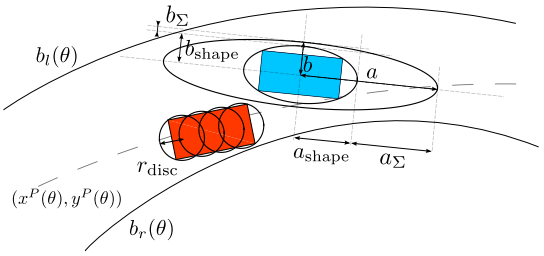


Fig. 5. Ego vehicle (red) approximated by circles of radius r_{disc} and other vehicle (blue) with shape- and uncertainty-ellipse corresponding to minimum occupancy probability p_ϵ. Road boundaries $b_l(\theta)$ on the left and right $b_r(\theta)$ of reference path $(x^P(\theta), y^P(\theta))$.

E. Avoidance of Dynamic Traffic Participants

For brevity we will refer to all traffic participants, such as vehicles, pedestrians, bicyclists, as vehicles. The ego-vehicle needs to plan its motion in dynamic, uncertain environments (DUE). Successful and efficient operation of autonomous systems in such environments requires reasoning about the future evolution and uncertainties of the states of the moving agents and obstacles. The evolution of trajectories of other vehicles can be reasonably predicted for a short time (typically less than a second) by just considering physical quantities such as vehicles estimated velocities, directions, and yaw rates, and estimating the growth of uncertainty by Gaussian noise over the inputs, e.g. in a LQG framework. The evolution over several seconds on the other hand is much stronger influenced by the intentions, motivations and goals of all traffic participants within the specific driving environment and thus a different representation is more suitable. While we will choose to carefully *model* the mean propagation based on vehicles following lanes and propagate the uncertainty accordingly, this should be seen as a means of providing inputs to our method. More elaborate estimation of posterior propagation and estimation is beyond the scope of this paper.

The stochastic dynamic programming problem that is associated with DUE planning [34], accounting for chance constraints that arise from the uncertain locations of the dynamic obstacles is computationally challenging for real-time applications. We will motivate the approximations necessary to render solving the DUE tractable in real-time in the following section.

We can check for collisions with chance constraints

$$P(C|\mathbf{z}_k, \mathbf{z}_k^i) < p_\epsilon, \tag{30}$$

where the general probability of collision is given by

$$P(C|\mathbf{z}_k, \mathbf{z}_k^i) = \int_{\mathcal{B}(\mathbf{z}_k)} \int_{\mathcal{B}^i(\mathbf{z}_k^i)} I_C(\mathbf{z}_k, \mathbf{z}_k^i) p(\mathbf{z}_k, \mathbf{z}_k^i)\, d\mathbf{z}_k\, d\mathbf{z}_k^i, \tag{31}$$

with the ego-vehicle's and vehicle i's states $\mathbf{z}_k$, $\mathbf{z}_k^i$, and footprints $\mathcal{B}(\mathbf{z}_k)$, $\mathcal{B}(\mathbf{z}_k^i)$ respectively, and indicator function

$$I_C(\mathbf{z}_k, \mathbf{z}_k^i) \coloneqq \begin{cases} 1, & \text{if } \mathbf{p}_k = \mathbf{p}_k^i \\ 0, & \text{otherwise.} \end{cases} \tag{32}$$

For the general case, including rectangular objects and general orientations, the probability of collision can only be estimated by Monte Carlo sampling which makes it unsuitable for use in real-time optimization.

Assuming conditional independence between vehicles, uncertainty only for the other vehicles, and no uncertainty over the host vehicle, simplifies Eq. (31) and yields the following constraint for encoding the chance constraint of Eq. (30):

$$\mathcal{B}(\mathbf{z}_k) \cap \mathcal{B}^i(\mathbf{z}_k^i, \mathbf{\Sigma}_k^i, p_\epsilon) = \emptyset \tag{33}$$

Where $\mathcal{B}(\mathbf{z}_k)$ is the host vehicle's footprint and $\mathcal{B}^i(\mathbf{z}_k^i, \mathbf{\Sigma}_k^i, p_\epsilon)$ the other vehicle's footprint with probability of more than p_ϵ. We will derive a closed form approximation for this constraint in the following.

For brevity, index i will be omitted in the further part of this section. The shapes of other traffic participants, such as vehicles, pedestrians, and bikes are approximated by a footprint encompassing ellipse $\mathcal{E}(a_{\text{shape}}, b_{\text{shape}})$ of orientation ϕ with semi-major axes a_{shape} and b_{shape} in longitudinal and lateral direction of the obstacle respectively, see Fig. 5. We take advantage of the availability of an analytic description of their occupied area to derive collision states that can be described in closed analytic form. The evolution of the obstacles' future trajectories are assumed to be known up to some uncertainty in the form of posterior distributions parameterized by mean trajectories $\mathbf{z}_{0:m}$ and uncertainty $\mathbf{\Sigma}_{0:m}$. For our instantiation we supply a model of the growth of uncertainty

$$\mathbf{\Sigma}_{k+1} = \min\left(\mathbf{\Sigma}_k + \mathbf{\Sigma}\Delta t_k, \mathbf{\Sigma}_{\max}\right), \tag{34}$$

of the vehicle's position with uncertainty $\mathbf{\Sigma}_k = \text{diag}[(\sigma_k^a)^2, (\sigma_k^b)^2]$ at time k. The growth of uncertainty is determined by $\mathbf{\Sigma} = \text{diag}[(\sigma^a)^2, (\sigma^b)^2]$. The uncertainty, i.e. covariance, in the vehicle's position is approximated to be aligned with the vehicle's heading and thus the principle axis of the encompassing ellipse (cf. Fig. 5). The uncertainty growth is bounded by $\mathbf{\Sigma}_{\max}$, especially in the lateral direction, to take the high likelihood of vehicles staying in their current lanes into account. While we have chosen this model of propagation of uncertainty for simplicity, other more complex models or learned estimates are feasible in this framework as well as long as the uncertainty can be approximately aligned with the vehicle's axis.

The level-set lines of the Gaussian $\mathcal{N}(0, \mathbf{\Sigma}_k)$ describing the position uncertainty of the other traffic participants at the level of p_ϵ form ellipses $\mathcal{L}(a_{\Sigma_k}, b_{\Sigma_k})$ with coefficients

$$\begin{bmatrix} a_{\Sigma_k} \\ b_{\Sigma_k} \end{bmatrix} = \begin{bmatrix} \sigma_k^a \\ \sigma_k^b \end{bmatrix} \left(-2\log(p_\epsilon 2\pi\sigma_k^a\sigma_k^b)\right)^{1/2}. \tag{35}$$

The Minkowski-sum $\oplus$ of the axis aligned uncertainty ellipse $\mathcal{L}(a_{\Sigma_k}, b_{\Sigma_k})$ and the shape ellipse $\mathcal{L}(a_{\text{shape}}, b_{\text{shape}})$ now allows us to receive an ellipse

$$\begin{aligned} \mathcal{B}^i(\mathbf{z}_k^i, \mathbf{\Sigma}_k^i, p_\epsilon) &\subset \mathcal{L}(a_{\Sigma_k}, b_{\Sigma_k}) \oplus \mathcal{L}(a_{\text{shape}}, b_{\text{shape}}) \\ &\subset \mathcal{L}(a_{\Sigma_k} + a_{\text{shape}}, b_{\Sigma_k} + b_{\text{shape}}), \end{aligned} \tag{36}$$

conservatively inscribing the vehicle's footprint up to an uncertainty p_ϵ at time k.

The axis alignment of the uncertainty ellipses to the car enables us to directly add the coefficients to the semi-major axes to find the obstacle's ellipse $\mathcal{B}(\mathbf{z}_k^i, \Sigma_k^i, p_\epsilon)$ with occupancy probability above the p_ϵ threshold.

The rectangular shape of the ego car is approximated by a set of 4 discs $\mathcal{R}^j(\mathbf{z}_k)$, $j \in \{1, \ldots, 4\}$, of radius r_{ego}

$$\mathcal{B}(\mathbf{z}_k) \subset \left(\bigcup_{j \in \{1,\ldots,4\}} \mathcal{R}^j(\mathbf{z}_k) \right) \tag{37}$$

chosen in a conservative manner, cf. Fig. 5, to completely enclose the vehicle.

It is necessary to employ discs instead of ellipses for the ego-vehicle, since the ego-vehicle and the other vehicles are not necessarily axis aligned and the Minkowski sum can not be easily derived for non-axis aligned ellipses in closed form. The Minkowski sum of the ego-car's discs and an ellipse on

the other hand has a closed form description. The collision constraint (33) can therefore be transformed to

$$\left(\bigcup_{j\in\{1,\dots,4\}} \mathcal{R}^j(\mathbf{z}_k) \right) \bigcap \mathcal{L}(a_{\Sigma_k} + a_{\text{shape}}, b_{\Sigma_k} + b_{\text{shape}}) = \emptyset, \tag{38}$$

and we can apply the Minkowski sum on the ego car's j-th discs $\mathcal{R}^j(\mathbf{z}_k)$ and the previously derived occupancy-ellipse to form analytic collision constraints

$$c_k^{\text{obst.,i}}(\mathbf{z}_k) = \left[\begin{array}{c} \Delta x_j \\ \Delta y_j \end{array} \right]^T R(\phi)^T \left[\begin{array}{cc} \frac{1}{a^2} & 0 \\ 0 & \frac{1}{b^2} \end{array} \right] R(\phi) \left[\begin{array}{c} \Delta x_j \\ \Delta y_j \end{array} \right] \Bigg|_{k,i} > 1, \quad \forall j \in \{1,\dots,4\} \tag{39}$$

where Δx_j, Δy_j are the distance of the ego vehicle's j-th disc to the center of the obstacle i at time k. $R(\phi)$ is the rotation matrix corresponding to the obstacle's heading, and

$$\left[\begin{array}{c} a \\ b \end{array} \right] = \left[\begin{array}{c} a_{\text{shape}} + a_{\Sigma_k} + r_{\text{disc}} \\ b_{\text{shape}} + b_{\Sigma_k} + r_{\text{disc}} \end{array} \right] \tag{40}$$

the semi-major axis of the resulting constraint-ellipse. We now have an analytic closed form constraint prohibiting collisions of probability higher than p_ϵ with other vehicles fulfilling the collision constraints Eq. (33) and Eq. (30) up to a conservative approximation.

For more elaborate methods of uncertainty propagation with necessity of e.g. sigma point transformations or Monte Carlo sampling, a conservative approximation of a Gaussian with principle axis aligned to the expected vehicle orientation, with new uncertainty ellipse coefficients $a_{\Sigma_k}, b_{\Sigma_k}$, can be computed. The uncertainty propagation needs to be completed only once pre-optimization and as a result does not affect optimization run-time as much. The only restriction lies in the alignment of the covariance along the vehicle's ellipse's semi-major axis. More generally, if the occupancy up to p_ϵ of other vehicles can be described by an ellipse the remainder of the method remains applicable without the need of the aforementioned restrictions such as axis alignment.

F. Minimal Intervention

It is our goal to follow the human input very closely and intervene only when deemed necessary. In the general case the intervention cost

$$J_{\text{h}}(\mathbf{z}_k, \mathbf{u}_k, \mathbf{u}_k^h) = \left[\{\mathbf{u}_k, \mathbf{z}_k\} - \mathbf{u}_k^h \right]^T K \left[\{\mathbf{u}_k, \mathbf{z}_k\} - \mathbf{u}_k^h \right] \tag{41}$$

with weights $K \in \mathbb{S}_{++}$ penalizes the deviation of the system's state or input from the human driver's predicted desired input $\mathbf{u}_k^h$. The function is adapted depending on which corresponding states $\mathbf{z}_k$ and control inputs $\mathbf{u}_k$ are observable from human inputs $\mathbf{u}_k^h$.

Instead of predicting future driver inputs $\mathbf{u}_0^h$ from a driver model like Shia et al. [20], we hold the human input constant $\mathbf{u}_k^h = \mathbf{u}_0^h \forall k$ and only account for the difference to the current human input. As we will see in the forthcoming section, Sec. IV-G, the impact of $J_{\text{h}}(\mathbf{z}_k, \mathbf{u}_k, \mathbf{u}_0^h)$ dominates the optimization in the short term but drops off quickly such that a prediction becomes irrelevant. The optimization plans trajectories sufficiently close to the short term intentions of the human driver, and subsequently follows the long-term goals with increasing time progress.

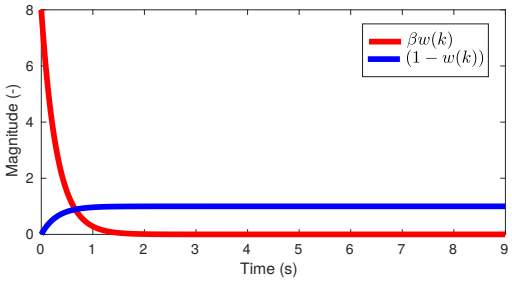


Fig. 6. Comparison of magnitude of minimal intervention cost J_h amplification ($\beta w(t_k)$), red, and trajectory cost J_h amplification ($1 - w(t_k)$), blue.

In our experimental setup we can only observe the driver's desired steering angle δ^h and acceleration $\dot{v}_x^h$, but not the steering speed $\dot{\delta}^h$. The minimal intervention cost accounting only for the deviation from the *current* human driver input $\mathbf{u}_0^h = [\delta_0^h, \dot{v}_{x,0}^h]^T$ becomes

$$J_{\text{h}}(\mathbf{z}_k, \mathbf{u}_k, \mathbf{u}_0^h) = \left[\begin{array}{c} \dot{v}_{x,k}^u - \dot{v}_{x,0}^h \\ \delta_k - \delta_0^h \end{array} \right]^T K \left[\begin{array}{c} \dot{v}_{x,k}^u - \dot{v}_{x,0}^h \\ \delta_k - \delta_0^h \end{array} \right]. \tag{42}$$

Nonetheless, the framework is general enough to take human input predictions, as well as other states such as steering velocity, acceleration or torque as human input into account, if observable.

G. Merging of Minimal Intervention and Trajectory Costs

We propose a linear, time-dependent combination of the cost of intervention J_h Eq. (42) and trajectory cost J_t Eq. (27)

$$J(\mathbf{z}_k, \mathbf{u}_k, \theta_k, \mathbf{u}_0^h) = \beta\omega(t_k) J_{\text{h}}(\mathbf{z}_k, \mathbf{u}_k, \mathbf{u}_0^h) + (1 - \omega(t_k))\, J_{\text{t}}(\mathbf{z}_k, \mathbf{u}_k, \theta_k), \tag{43}$$

with the goal of decreasing dependence on accurate prediction of future driver commands $\mathbf{u}_{0:m}^h$ and instead only relying on the current human input $\mathbf{u}_0^h$.

Weights $\beta \in \mathbb{R}_+$ and an exponential decay function $w(t_k) = \exp(-\alpha t_k)$, $\alpha \in \mathbb{R}_+$, are chosen to increase the impact of the human input in the short-term, cf. Fig 6. $w(t_k)$, the relative weight of J_h is chosen to drop off to 10% of it's initial value after only 0.5s. Therefore, and because of a large β, the optimization is dominated by the minimal intervention cost J_h in the short term to make the system very responsive to current human inputs. With increasing time it will more and more rely on J_t, while J_h becomes increasingly more irrelevant. While the system's output trajectory further in the future might arguably not coincide with the human driver's anticipated trajectory, it will eventually *snap* into the correct human long term intention with progression of time. A deviation of the system will only become noticeable if safety constraints are active and the system will be perceived as inactive to the human driver otherwise. A study on driving in shared safety systems [25] underlines the importance: Human

drivers approved of a surprisingly high amount of intervention if their own high-level goals were achieved. The authors in [25] argue that intervention is not perceived as such if it does not follow adversary goals.

Since only the trajectory cost affects the long-term, all benefits noted in Sec. IV-B, especially guidance for the optimizer, become more prevalent for future planning steps.

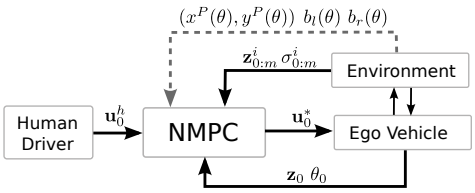


Fig. 7. Control scheme of the NMPC

H. Optimization

We formulate the optimization problem with the aforementioned state-, dynamics-, path-, and obstacle constraints and form the following constrained nonlinear optimization problem:

$$\mathbf{u}^*_{0:m-1} = \underset{\mathbf{u}_{0:m-1}}{\arg\min} \sum_{k=0}^{m} J(\mathbf{z}_k, \mathbf{u}_k, \theta_k, \mathbf{u}_0^h)\Delta t_k \tag{44}$$

$$\text{s.t. } \mathbf{z}_{k+1} = f(\mathbf{z}_k, \mathbf{u}_k) \tag{45}$$

$$\theta_{k+1} = \theta_k + v_k \Delta t_k \tag{17}$$

$$\mathbf{z}_{\min} < \mathbf{z}_k < \mathbf{z}_{\max} \tag{46}$$

$$\mathbf{u}_{\min} < \mathbf{u}_k < \mathbf{u}_{\max} \tag{47}$$

$$|\phi_k - \phi^P(\theta_k)| < \Delta\phi_{\max} \tag{29}$$

$$|v_x \dot{\phi_k}| < (v_x \dot{\phi})_{\max} \tag{4}$$

$$F_\alpha = \sqrt{F_{\alpha x}^2 + F_{\alpha y}^2} \leq \mu_{\alpha,\max} F_{\alpha z}, \;\; \alpha \in \{f, r\} \tag{13}$$

$$b_l(\theta_k) + w(\Delta\phi_k) \leq d(\mathbf{z}_k, \theta_k) \leq b_r(\theta_k) - w(\Delta\phi_k) \tag{28}$$

$$c_k^{\text{obstacle,i}}(\mathbf{z}_k) > 1, \;\; i = \{1, \ldots, n\} \quad \forall k \in \{0, \ldots, m\}. \tag{39}$$

Constraint (4) is active for the kinematic model only, while (13) is only active for the dynamical model. At initialization the path $(x^P(\theta), y^P(\theta))$ and boundaries $b_l(\theta)$ and $b_r(\theta)$ are given by the road and static obstacles, cf. Fig. 7 and Alg. 1. The boundaries are updated once new information about roads or static obstacles become available. At the beginning of each control loop the initial states $\mathbf{z}_0$, θ_0, human control input $\mathbf{u}_0^h$, and predictions of other traffic participants mean $\mathbf{z}^i_{0:m}$ and uncertainties $\Sigma^i_{0:m}$ are provided to the NMPC and the corresponding constraint ellipses are computed. Subsequently Eq. (44) is solved and the resulting optimal control $\mathbf{u}_0^*$ is executed by the system. The system consequently returns to the beginning of the loop, see Fig. 7 and Alg. 1. The optimization problem Eq. (44) is solved by a Primal-Dual Interior Point solver generated by FORCES Pro [26].

Algorithm 1 Summary of NMPC control flow

Sense environment $\mathcal{E}$ including static obstacles $\mathcal{O}$ and dynamic obstacles $\mathbf{z}_0^i$, $a^i_{\text{shape}}, b^i_{\text{shape}}$, $i \in \{1, \ldots, n\}$;
Initialize $(x^P(\theta), y^P(\theta))$, the path spline of the ego vehicle from the road network;
Compute $b_l(\theta)$, $b_r(\theta)$, the path-boundary splines including road constraints and static obstacles $\mathcal{O}$;
loop
  Sense env. $\mathcal{E}$, i.e. $\mathcal{O}$, $\mathbf{z}_0^i$, $a^i_{\text{shape}}, b^i_{\text{shape}}$, $i \in \{1, \ldots, n\}$;
  if Static environment changed **then**
    Update $\left(x^P(\theta), y^P(\theta)\right)$;
    Compute $b_l(\theta)$, $b_r(\theta)$;
  end if
  for $i \in \{1, \ldots, n\}$ **do**;
    Predict $\mathbf{z}^i_{1:m}$;
    Propagate $\Sigma^i_{0:m}$; ▷ Eq. (34)
    Compute $(a^i_{0:m}, b^i_{0:m})$; ▷ Eq. (40)
  end for
  Find θ_0, ego vehicle path abscissa; ▷ Eq. (18)
  Measure $\mathbf{z}_0$, the current ego state;
  Measure $\mathbf{u}_0^h$, the current human driver input;
  Compute $\mathbf{u}_0^*$; ▷ Eq. (44)
  Apply $\mathbf{u}_0^*$ to system;
end loop

I. Technical Discussion

The described method guarantees dynamic collision avoidance while staying within the limits of the road up to the time horizon, under the conditions that (a) the solver is able to find a solution, and (b) knowledge of the road, static obstacles, dynamic obstacles on the road, and a prediction of the mean states of the dynamic obstacles up to a sufficient uncertainty are available. Since this work does not address recursive feasibility (a) can not always be guaranteed.

1) Quality of the trajectory: The optimization problem is non-convex and, e.g. in the event of deciding to overtake a vehicle on the left or right, can become of combinatorial nature. Although the Primal-Dual Interior Point method solves for a local solution, we have achieved good results even for poor initialization for four reasons:

1) The local solution trajectory will *snap* into the human's desired trajectory eventually as time progresses. Only the current control $\mathbf{u}_0$ is executed and since the current human inputs $\mathbf{u}_0^h$ dominate the cost function, as discussed in Sec. IV-G, it is acceptable to currently execute a local solution.
2) Any local solution is safe. While not necessarily optimal in the sense of minimal intervention it will abide by the safety constraints.
3) Increased static and dynamic regularization parameters yield robust optimization in practice.
4) The J_{MPCC} cost term shapes the trajectory optimization process closer to convex by encouraging progress along the path and creating a cost-basin by penalizing large lateral offsets to the path.

Executing the NMPC, including solving a nonlinear non-

TABLE II
COST WEIGHTS

Function	Parameter	Value
MPCC	$Q = \text{diag}(q_{\text{long}}, q_{\text{lat}})$	$\text{diag}(1.0, 1.0)$
Progress	ρ	5.0
Control	$R = \text{diag}(r_{acc}, r_{\dot{\delta}})$	$\text{diag}(1.0, 1.0)$
Yaw rate	α	0.1
Minimal intervention	$K = \text{diag}(k_{\dot{v}}, k_{\delta})$	$\text{diag}(1.0, 2.0)$
Scaling of intervention	β	500

convex optimization problem, has a number of general drawbacks such as uncertain convergence, potentially unbounded runtime, and the lack of guarantees of optimality. In future work a high-level trajectory planner, solving the combinatorial problem, could be applied to yield a meaningful initialization to ensure global optimization.

2) Number of Variables: Per stage 1 state originates from the path integrator θ_k, 2 inputs, and 5 (7) states $\mathbf{z}_k$ for the kinematic (dynamical) model, thus 8 (10) variables exist. The NMPC plans over 50 stages, including 8 (10) variables per stage, and thus 394 (492) variables in total excluding the initial states for the kinematic (dynamical) model are needed.

3) Number of Constraints: Including the system dynamics and path progress evolution in total 6 (8) equality constraints need to be respected. 14 (16) inequality constraints result from state, input, velocity-yaw-rate, (friction), heading-deviation, and road boundary constraints. $4n$ additional inequality constraints originate from n dynamic obstacles in the environment and the 4 circle approximation of the ego vehicle. In total $20\,(24) + 4n$ constraints are specified per stage, while depending on the solver only a subset of those are active.

V. RESULTS

We evaluate the capabilities of our approach in a variety of simulated scenarios. The human driver controls a physical steering wheel and pedals, see Fig. 8, which generate the desired inputs $\mathbf{u}_0^h = [\delta_0^h, \dot{v}_{x,0}^h]$, i.e. steering angle δ_0^h and acceleration $\dot{v}_{x,0}^h$. The human inputs are then processed in the NMPC formulation to guarantee safe motion. The reference path and the road boundaries b_l and b_r are designed to fit the road network. We adopt a variable step size approach in all scenarios, and for all motion models, to increase the time horizon of the planner without sacrificing computation cost. During the first 10 steps we employ $\Delta t_k = 0.1s$ and $\Delta t_k = 0.2s$ for the remaining 40 steps, resulting in a planning horizon of nearly $9s$. During all experiments the cost function's weights remained unchanged and are displayed in Tab. II.

A. Left Turn Across Traffic, Merging, and Overtaking

In this challenging left turn cross traffic scenario, see Fig. 1 and Fig. 9, the ego-vehicle intends to merge into the oncoming traffic while avoiding collisions with cross traffic. Afterwards, the driver attempts to overtake a vehicle while traffic obstructs the maneuver. For this challenging scenario with multiple dynamic obstacles and a decision-making component the simpler kinematic model was employed.

Fig. 8. Virtual driving setup with steering wheel and pedals.

We have evaluated the method in a set of 100 randomly generated scenarios. In these scenarios the initial positions, trajectories including acceleration and velocity profiles, of all other traffic participants were randomly generated. The Parallel Autonomy system was able to ensure safety at all times, although we purposefully caused unsafe human driver inputs which would have resulted in crashes without the proposed system.

In the following we will present two representative examples for two different human driving styles, i.e. an aggressive and a calm driver. We define the total intervention as a direct measure to compare the NMPC's behavior subject to different scenarios and driving styles:

$$\text{Intervention} := \frac{100}{2}\left(\frac{|\dot{v}_{x,0} - \dot{v}_{x,0}^h|}{2\dot{v}_{x,\max}} + \frac{|\delta_0 - \delta_0^h|}{2\delta_{\max}}\right) \ [\%]. \tag{48}$$

We measure the amount of total intervention as the sum of deviation from the human input normalized with the maximum possible inputs scaled to $\%$.

1) Aggressive Driver: In the first case, cf. Fig. 9, an aggressive driver nearly collides with the right road boundary even before entering the intersection ①, which is prevented by counter-steering of the autonomous system since the right road boundary constraint becomes active. Then, the driver tries to accelerate into the intersection ②, although other vehicles are just passing by, resulting in a near collision. Our system brings the ego vehicle to a full stop, lets the other vehicles pass and then proceeds by letting the driver merge into the traffic when a large enough gap appears. Here the dynamic obstacle constraints prevented a collision and influenced the merging behavior. To achieve this high-level of reasoning and scheduling both a long planning horizon as well as low-level control on a trajectory basis combined with high replanning frequency to quickly account for changes in the environment are necessary.

At ③ the driver approaches a preceding vehicle with high relative speed and tries to collide by accelerating even further. Our system brakes the ego vehicle, first with gradually increasing effort and then with maximum acceleration $a_{\max}$, and allows an overtaking maneuver once the oncoming traffic has passed. At ④ the driver erratically tries to break through the right road boundary, which is prohibited by our system and the ego vehicle is able to continue driving on the road in a safe manner. In all these cases the system can guard the human driver from actually causing any harm to himself and

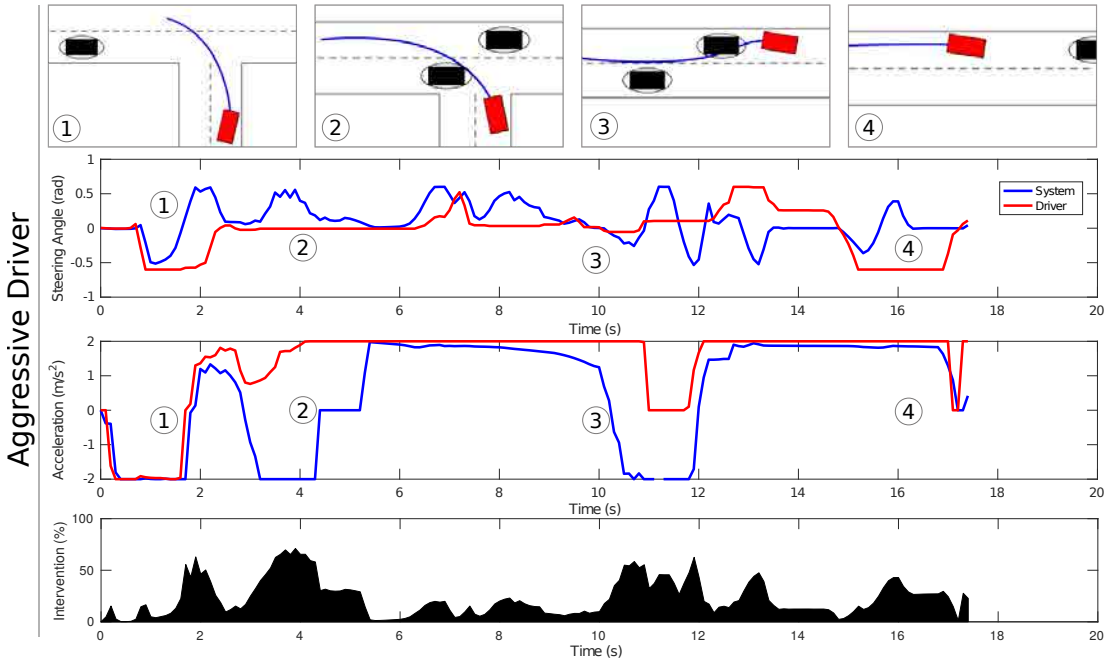


Fig. 9. *Aggressive left turn with traffic*: The system's steering angle and acceleration are displayed in blue, the human input in red. Snapshots of the current scenes at specific time-stamps are displayed above the acceleration and steering plots: The ego vehicle in red, the MPC planned path in blue. All other vehicles in black. An aggressive driver causes multiple critical situations where the system is forced to intervene to large amounts to keep the vehicle in a safe state. Large deviation from the driver's desired acceleration and steering wheel angle to the actual system output are observable. E.g. collision at time (2) is prohibited by strong braking.

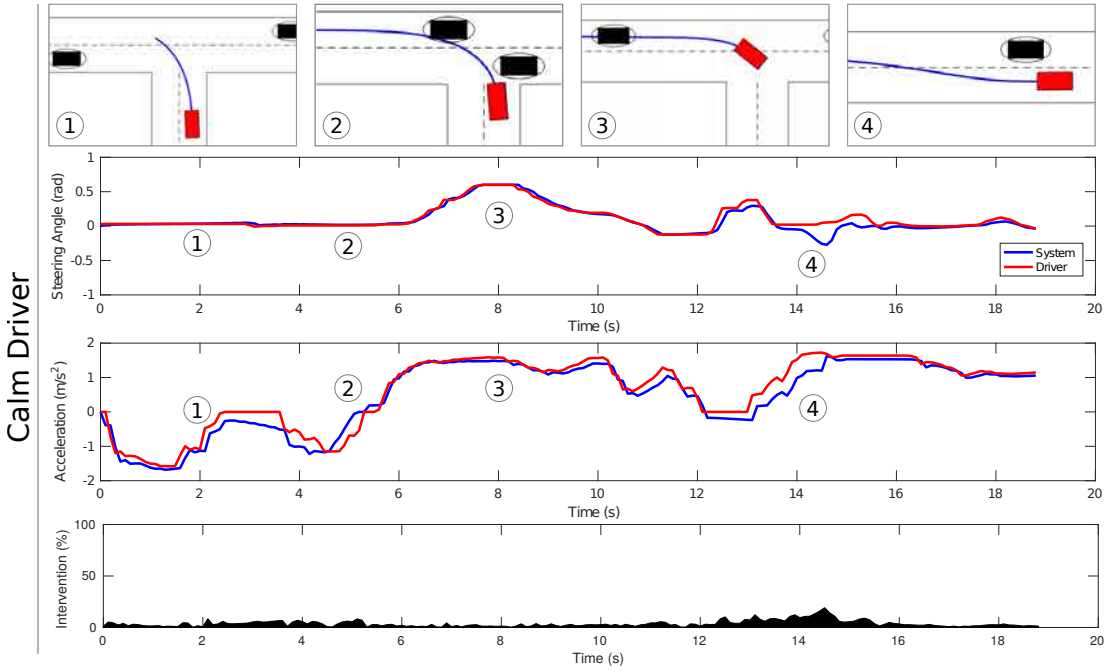


Fig. 10. *Normal left turn with traffic*: System output stays close to the desired human acceleration and steering wheel angle. An exception appears at (4) where the driver is not counter steering enough to prohibit a predicted collision with the left road boundary.

others.

Due to the severity of the scenario a maximum intervention of 71% and 23% on average was necessary to assure safety.

2) Calm Driver: The opposite spectrum of how our method reacts is shown in Fig. 10: A fairly good driver experiences the same previous scenario. We observe that if the inputs from the human driver are deemed safe, barely any difference between human and system inputs occurs. The system thus minimizes intervention if no critical situations occur. Since steering the vehicle with steering wheel and pedals in simulation is not an easy task, due to the lack of feedback, the human driver did not break sufficiently at (2) and misses to counter steer

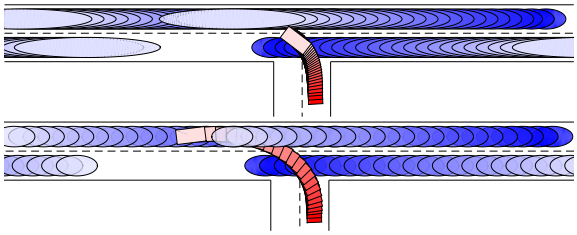


Fig. 11. Comparison of NMPC *plans* with uncertainty estimate (top) and without (bottom) shown by the ellipses representing their occupancy probability threshold. Predicted future states are shown in fading colors $0.4s$ apart over a horizon of $9s$.

during a lane change maneuver ④. Here the system applied the slight *nudging* behavior to carefully guide the driver away from the imminent collision with the left road boundary. Even in this situation the maximum intervention did not exceed 16% and was only 3% on average for the whole scenario, which underlines the functionality of the minimal intervention principle.

B. Impact of Uncertainty

Taking the uncertainty in the prediction of other vehicles into account is important, since future states can deviate substantially from the expectation. In the case of neglecting uncertainties the planned behavior can be more aggressive and is given larger leeway in the constraints. See Fig. 11-bottom, where the vehicle is allowed to merge into the lane in front of a second vehicle. Taking future obstacles' uncertainty growth into account, cf. Fig. 11-top, results in more conservative behavior and the ego vehicle is prohibited from merging in front of the oncoming vehicle. We discussed here the planned behavior of the NMPC, not the actually observed behavior. Since the control loop runs at more than $10Hz$, frequently updating the planned trajectory, the actually executed controls will be less conservative since they adapt to new observations. It will be possible to actually observe the true positions and velocities of the other vehicles over time and thus replan with lower uncertainty. Nonetheless, we have shown the impact of uncertainty aware planning by including chance constraints into the optimization.

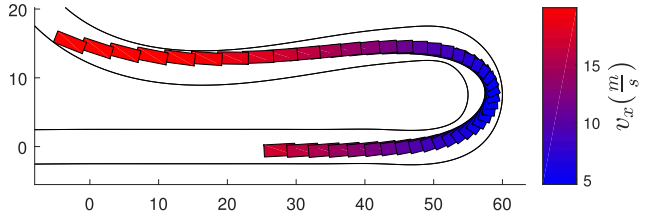
C. Snow Race Track

TABLE III
VEHICLE SPECIFICATIONS

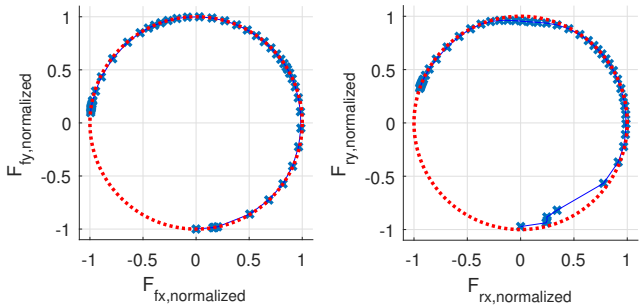
Parameter	Value
l_f, l_r, w, h	$1.2, 1.5, 1.76, 0.9$ (m)
m, I_z	$1400kg$, $3000kg/m^2$
$B_\alpha, C_\alpha, D_\alpha$ (snow)	$5, 2, 0.3$

In the following scenarios we will showcase the capabilities of the dynamical vehicle model on a snow surface in fast turns and static obstacle avoidance on a race track. Vehicle and tire specifications are stated in Table. III.

1) Sharp turn: In this scenario, cf. Fig. 12(a), the vehicle enters a sharp left turn on a race track. The current human inputs would cause the vehicle to quickly leave the road at high speed. The controller brakes the vehicle to a safe speed complying with the friction constraints, see Fig. 12(b), then accelerates at the exit of the turn to maximize progress while always respecting the road's limits. The planned trajectory



(a) NMPC *plan* of vehicle, where vehicle poses are $0.2s$ apart for a horizon of $9s$.



(b) The friction constraints for both tires are visualized as normalized components $F_{\alpha\beta,\text{normalized}} = F_{\alpha\beta}/(\mu_{\alpha,\max} F_{\alpha z})$.

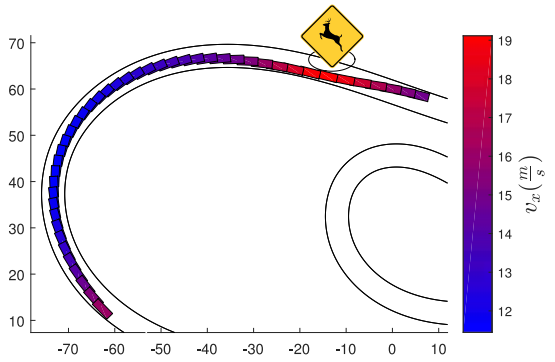
Fig. 12. *Sharp turn*: Output of NMPC plans to decelerate into the sharp corner to comply with friction constraints.

shows similarities to a racing line during high-speed cornering. This behavior shows the advantage of longitudinal and lateral control; without deceleration the vehicle would not have been able to complete the turn, cf. Fig. 12(a). The plan maintains a smooth acceleration profile during the turn and maximizes the use of available forces, cf. Fig. 12(b), while abiding by the friction constraint.

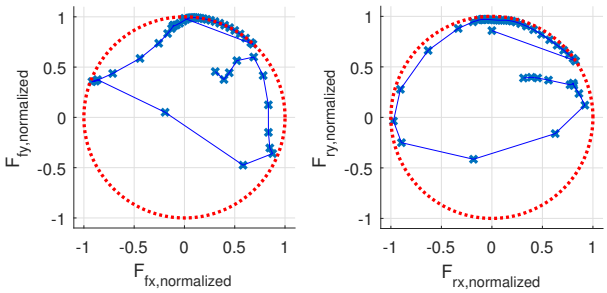
2) Sudden appearance of obstacle: A suddenly appearing static obstacle represented by an ellipse in the driving-path of the vehicle, e.g. a stationary deer, needs to be avoided, cf. 13(a). After replanning a new trajectory the vehicle is able to swerve to the left side of the road to avoid the static obstacle in its previous driving path. The optimization does not have any incentive to slow down in the vicinity of obstacles and thus continues to accelerate to increase progress along the road until it brakes for the imminent left corner.

D. Computation Time

The NMPC solve times collected during several runs on a single core CPU are displayed in Fig. 14. For the kinematic model we observed a strong influence of the complexity of the scenario on the computation time. In the case of no dynamic obstacles we observed solve-times of less than $20ms$ even for a challenging race track with many tight turns, forcing the NMPC to intervene and decelerate due to velocity-yaw-rate



(a) NMPC *plan* of vehicle avoiding suddenly appearing obstacle, where vehicle poses are $0.2s$ apart for a horizon of $9s$.



(b) The friction constraints remain valid during the avoidance maneuver.

Fig. 13. *Sudden appearance of obstacle*: The vehicle is able to successfully avoid the suddenly appearing obstacle.

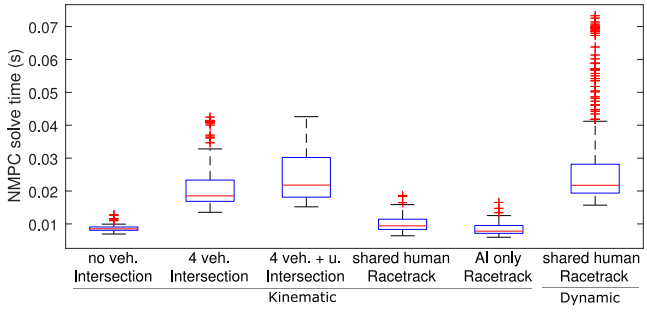


Fig. 14. Computation times for the different human in the loop scenarios and varying number of other traffic participants, and high uncertainty. Finally the case of AI only, without the human in the loop. On the far right is the dynamic vehicle model in a static environment. Results were computed on a single core of an AMD Ryzen 7 1700X @3.4Ghz.

constraints. In cases where the system needs to nudge into tight gaps while simultaneously deciding whether a subsequent overtaking maneuver is feasible, computation times can reach up to $50ms$ in exceptional cases.

The dynamical vehicle model is surprisingly fast on average but exhibits worst case run-times of nearly $75ms$. While we achieved robust performance and convergence for all tests in static environments, for some cases in dynamic environments no optimizer could be found and we therefore exclude the dynamical model from those use cases.

In summary, our system was able to reach the goal replanning frequency of $10Hz$ at all times for the kinematic vehicle model in complex dynamic environments, and the dynamical model in simpler and static environments.

VI. Conclusion

In this work we presented a NMPC that minimizes deviation from the human input while ensuring safety according to our proposed general Parallel Autonomy control framework. We have shown the increased functionality compared to other approaches in complex and more realistic driving scenarios. The approach is capable of reasoning over long time horizons of more than $9s$ in real-time, i.e. more than $10Hz$, while maintaining close to the human input without a necessary prediction layer for human intention.

We have shown our method to work with a kinematic model in challenging and highly complex dynamic environments, and a dynamical model in static environments.

Future work will try to reduce general limitations of NMPC, such as uncertain convergence and lack of a guarantee of optimality by initialization in a correct homotopy class, or by exploring strategies on how to deal with failure cases where no solution can be found, or to extend the presented method to achieve provable safety beyond the planning horizon by ensuring recursive feasibility. Additional tests will also enclose an inference framework to gain more elaborate predictions of other traffic participants, future trajectories and their uncertainty propagation.

Instead of nudging the driver in the correct direction, it is possible to not intervene at all if the driver is doing well. The impact of the different methodologies on human drivers will be studied in the future. Furthermore, the proposed receding horizon planner also applies to fully autonomous vehicles if the minimal intervention cost is excluded and future experiments will show the functionality.

References

[1] W. Schwarting, J. Alonso-Mora, L. Paull, S. Karaman, and D. Rus, "Parallel Autonomy in Automated Vehicles: Safe Motion Generation with Minimal Intervention," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*.
[2] ASIRT, "Association for Safe International Road Travel 2016."
[3] National Safety Council (NSC), "Motor Vehicle Fatality Estimates 2012-2015."
[4] NHTSA, "Traffic Safety Facts 2015."
[5] M. Nohmi, N. Fujikura, C. Ueda, and E. Toyota, "Automatic braking or acceleration control system for a vehicle," Jan. 3 1978, US Patent 4,066,230.
[6] E. Dagan, O. Mano, G. P. Stein, and A. Shashua, "Forward collision warning with a single camera ," in *IEEE Intelligent Vehicles Symposium, 2004*, June 2004, pp. 37–42.
[7] M. Forghani, J. M. McNew, D. Hoehener, and D. D. Vecchio, "Design of driver-assist systems under probabilistic safety specifications near stop signs," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 1, pp. 43–53, Jan 2016.
[8] H. Ahn, A. Rizzi, A. Colombo, and D. Del Vecchio, "Robust supervisors for intersection collision avoidance in the presence of uncontrolled vehicles," *arXiv preprint arXiv:1603.03916*, 2016.
[9] D. Bresch-Pietri and D. D. Vecchio, "Estimation for decentralized safety control under communication delay and measurement uncertainty," *Automatica*, vol. 62, pp. 292 – 303, 2015.
[10] D. Hoehener, G. Huang, and D. D. Vecchio, "Design of a lane departure driver-assist system under safety specifications," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, Dec 2016, pp. 2468–2474.
[11] T. Fraichard and H. Asama, "Inevitable collision states. A step towards safer robots?" *Advanced Robotics*, vol. 18, no. 10, pp. 1001–1024, 2004.
[12] A. Bautin, L. Martinez-Gomez, and T. Fraichard, "Inevitable Collision States: A probabilistic perspective," in *2010 IEEE International Conference on Robotics and Automation*, May 2010, pp. 4022–4027.

[13] D. Althoff, M. Althoff, D. Wollherr, and M. Buss, "Probabilistic collision state checker for crowded environments," in *2010 IEEE International Conference on Robotics and Automation*, May 2010, pp. 1492–1498.

[14] N. Chan, J. Kuffner, and M. Zucker, "Improved Motion Planning Speed and Safety using Regions of Inevitable Collision," in *17th CISM-IFToMM Symposium on Robot Design, Dynamics, and Control (RoManSy'08)*, July 2008.

[15] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin, "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games," *IEEE Transactions on Automatic Control*, vol. 50, no. 7, pp. 947–957, July 2005.

[16] Y. Chen, H. Peng, and J. Grizzle, "Obstacle Avoidance for Low-Speed Autonomous Vehicles With Barrier Function," *IEEE Transactions on Control Systems Technology*, vol. PP, no. 99, pp. 1–13, 2017.

[17] S. J. Anderson, S. B. Karumanchi, and K. Iagnemma, "Constraint-based planning and control for safe, semi-autonomous operation of vehicles ," in *2012 IEEE Intelligent Vehicles Symposium*, June 2012, pp. 383–388.

[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," *International Journal of Vehicle Autonomous Systems*, vol. 8, no. 2-4, pp. 190–216, 2010.

[19] A. Constantin, J. Park, and K. Iagnemma, "A margin–based approach to vehicle threat assessment," *International Journal of Vehicle Autonomous Systems*, vol. 12, no. 4, pp. 384–411, 2014.

[20] V. A. Shia, Y. Gao, R. Vasudevan, K. D. Campbell, T. Lin, F. Borrelli, and R. Bajcsy, "Semiautonomous vehicular control using driver modeling," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 6, pp. 2696–2709, Dec 2014.

[21] Y. Gao, A. Gray, A. Carvalho, H. E. Tseng, and F. Borrelli, "Robust nonlinear predictive control for semiautonomous ground vehicles," in *2014 American Control Conference*, June 2014, pp. 4913–4918.

[22] S. M. Erlien, S. Fujita, and J. C. Gerdes, "Shared steering control using safe envelopes for obstacle avoidance and vehicle stability," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 2, pp. 441–451, Feb 2016.

[23] J. Alonso-Mora, P. Gohl, S. Watson, R. Siegwart, and P. Beardsley, "Shared control of autonomous vehicles based on velocity space optimization ," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 1639–1645.

[24] J. Storms, K. Chen, and D. Tilbury, "A shared control method for obstacle avoidance with mobile robots and its interaction with communication delay," *The International Journal of Robotics Research*, 2017.

[25] S. J. Anderson, S. B. Karumanchi, K. Iagnemma, and J. M. Walker, "The intelligent copilot: A constraint-based approach to shared-adaptive control of ground vehicles," *Intelligent Transportation Systems Magazine, IEEE*, vol. 5, no. 2, pp. 45–54, 2013.

[26] A. Domahidi and J. Jerez, "FORCES Pro," embotech GmbH (http://embotech.com/FORCES-Pro), Jul. 2014.

[27] T. Faulwasser, B. Kern, and R. Findeisen, "Model predictive path-following for constrained nonlinear systems ," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, Dec 2009, pp. 8642–8647.

[28] D. Lam, C. Manzie, and M. C. Good, "Model predictive contouring control for biaxial systems," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 2, pp. 552–559, March 2013.

[29] A. Liniger, A. Domahidi, and M. Morari, "Optimization-based autonomous racing of 1: 43 scale RC cars ," *Optimal Control Applications and Methods*, vol. 36, no. 5, pp. 628–647, 2015.

[30] F. Havlak and M. Campbell, "Discrete and Continuous, Probabilistic Anticipation for Autonomous Robots in Urban Environments," *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 461–474, April 2014.

[31] J. hwan Jeon, R. V. Cowlagi, S. C. Peters, S. Karaman, E. Frazzoli, P. Tsiotras, and K. Iagnemma, "Optimal motion planning with the half-car dynamical model for autonomous high-speed driving," in *2013 American Control Conference*, June 2013, pp. 188–193.

[32] E. Velenis, P. Tsiotras, and J. Lu, "Optimality Properties and Driver Input Parameterization for Trail-braking Cornering," *European Journal of Control*, vol. 14, no. 4, pp. 308 – 320, 2008.

[33] E. Bakker, L. Nyborg, and H. B. Pacejka, "Tyre Modelling for Use in Vehicle Dynamics Studies," in *SAE Technical Paper*. SAE International, 02 1987.

[34] N. E. D. Toit and J. W. Burdick, "Robot Motion Planning in Dynamic, Uncertain Environments," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 101–115, Feb 2012.

Wilko Schwarting Wilko Schwarting received the BSc in Mechanical Engineering and MSc in Robotics, Systems and Control from ETH Zurich, Switzerland, in 2014 and 2016 respectively. He is currently pursuing the Ph.D. degree in Electrical Engineering and Computer Science at the Computer Science and Artificial Intelligence Laboratory (CSAIL) at MIT. Wilko Schwarting received the Best Paper Award at IEEE ITSC 2014 in Qingdao.

Javier Alonso-Mora Javier Alonso-Mora is an assistant professor at the Delft University of Technology, Netherlands. Previously he was a Postdoctoral Associate at the Computer Science and Artificial Intelligence Laboratory (CSAIL) of MIT, USA. He received the M.Sc. and Ph.D. degrees in Robotics from ETH Zurich, Switzerland, in 2010 and 2014 respectively. From 2009 to 2014, he was also affiliated with Disney Research Zurich. His main research interest is in motion planning and control of autonomous multi-robot systems. Building towards the smart cities of the future, he develops constrained optimization algorithms that span across multiple applications, such as self-driving cars, automated factories, aerial vehicles and intelligent transportation systems.

Liam Paull Liam Paull is an assistant professor in the Département d'informatique et de recherche opérationnelle (DIRO) at the Université de Montréal and an associate member of the Montreal Institute for Learning Algorithms (MILA). Previously, he was a research scientist at MIT, where he helped lead the Toyota Research Institute funded parallel autonomy driving project. He also completed a Postdoctoral Associate at MIT where he studied resource-constrained perception systems with applications to underwater robotics. He completed his PhD in 2013 at the University of New Brunswick on marine robot planning and navigation. His expertise is in robot perception and the modeling of uncertainty.

Sertac Karaman Sertac Karaman is an Associate Professor of Aeronautics and Astronautics at the Massachusetts Institute of Technology (since Fall 2012). He has obtained B.S. degrees in mechanical engineering and in computer engineering from the Istanbul Technical University, Turkey, in 2007; an S.M. degree in mechanical engineering from MIT in 2009; and a Ph.D. degree in electrical engineering and computer science also from MIT in 2012. His research interests lie in the broad areas of robotics and control theory. In particular, he studies the applications of probability theory, stochastic processes, stochastic geometry, formal methods, and optimization for the design and analysis of high-performance cyber-physical systems. The application areas of his research include driverless cars, unmanned aerial vehicles, distributed aerial surveillance systems, air traffic control, certification and verification of control systems software, and many others. He is the recipient of an Army Research Office Young Investigator Award in 2015, National Science Foundation Faculty Career Development (CAREER) Award in 2014, AIAA Wright Brothers Graduate Award in 2012, and an NVIDIA Fellowship in 2011.

Daniela Rus Daniela Rus is the Andrew (1956) and Erna Viterbi Professor of Electrical Engineering and Computer Science and Director of the Computer Science and Artificial Intelligence Laboratory (CSAIL) at MIT. Rus's research interests are in robotics, mobile computing, and big data. The key focus of her research is to develop the science of networked/distributed/collaborative robotics, by asking: how can many machines collaborate to achieve a common goal? Rus is a Class of 2002 MacArthur Fellow, a fellow of ACM, AAAI and IEEE, and a member of the National Academy of Engineering. She earned her PhD in Computer Science from Cornell University. Prior to joining MIT, Rus was a professor in the Computer Science Department at Dartmouth College.