Model Predictive Control for Safe Autonomous Driving Applications

Ivo Batkovic, Mario Zanon, and Paolo Falcone

Abstract Although Model Predictive Control is widely used in motion planning and control for autonomous driving applications, accommodating closed-loop stability w.r.t. an *arbitrary* reference trajectory and avoidance of pop-up or moving obstacles is still an open problem.

While it is well-known how to design a closed-loop stable MPC w.r.t. a reference trajectory that satisfies the system dynamics, this chapter discusses how to guarantee stability of a vehicle motion planner and controller when a user-provided *arbitrary* reference is used. Furthermore, the proposed MPC scheme enables recursive collision-avoidance constraint satisfaction in the presence of pop-up or moving obstacles (e.g., pedestrians, cyclists, human-driven vehicles), provided that their predicted future motion trajectory is available together with some uncertainty bound and satisfies some mild requirement.

The proposed motion planner and controller is demonstrated through simulations.

Ivo Batkovic
Zenseact AB, Lindholmspiren 2, 417 56 Göteborg, e-mail: ivo.batkovic@zenseact.com
Chalmers University of Technology, Chalmersplatsen 4, 412 96 Göteborg, e-mail: ivo.batkovic@chalmers.se

Mario Zanon
IMT School for Advanced Studies Lucca, Piazza S.Francesco, 19, 55100 Lucca LU, Italy e-mail: mario.zanon@imtlucca.it

Paolo Falcone
Engineering Department Università di Modena e Reggio Emilia, Via Università, 4, 41121 Modena MO, Italy e-mail: paolo.falcone@unimore.it
Chalmers University of Technology, Chalmersplatsen 4, 412 96 Göteborg, e-mail: falcone@chalmers.se

1 Introduction

In order to fully deploy highly automated driving technologies, vehicles need to be able not only to reliably sense their surrounding environment, but also to *safely* interact with it. To that end, challenging problems need to be solved that span from robust and reliable sensors design (e.g., cameras, lidars, radars, GPS, HD-maps) to the development of robust perception and motion planning and control algorithms. While *safe* autonomous driving in complex environments still remains an open problem, research is progressing in the fields of localization [46, 47], object detection and tracking [10, 15, 33], and planning [23, 41, 45], to move beyond the current state of the art.

In this chapter, we focus on the vehicle motion planning and control problems for autonomous driving applications. We build our results on the Model Predictive Control (MPC) technique, as it has been proven to be a convenient design tool for self-driving applications including, e.g., optimal coordination [11, 12, 32], energy consumption minimization [31, 48, 52], and planning [5, 6, 13, 14, 24, 38, 29, 40]. The theory of MPC is equipped with well-known, well-established tools to enforce closed-loop stability w.r.t. a reference trajectory, while ensuring constraint satisfaction [9, 44]. However, these results build upon assumptions that can be challenging, or impossible, to satisfy in practical autonomous driving applications.

Such challenges include: a) enforcing closed-loop stability w.r.t. reference trajectories that do not satisfy the system dynamics; and b) providing recursive feasibility guarantees in uncertain environments (e.g., in presence of pop-up or moving obstacles). Indeed, if the MPC controller is provided with a reference trajectory that *does not* satisfy the system dynamics, then the well-known results for asymptotic stability of the closed-loop system do no longer hold. Furthermore, in an autonomous driving setting, the vehicle needs to interact with other road users that may appear almost anywhere and at any point in time within the sensor range. Ensuring that the vehicle motion planner can *persistently* be able to avoid collisions with other road users *at all future times* in such uncertain settings still remains an open problem. To address a) we resort to Input-to-State Stability (ISS) analysis to prove closed-loop stability also when *infeasible* (in the sense that they do not satisfy the system dynamics) references are used; and to address b) we provide a scheme which enables recursive collision-avoidance constraint satisfaction in the presence of pop-up or moving obstacles, provided that the uncertainty on their predicted motion is *not growing* as new information is available about the environment and the road users therein.

This chapter is structured as follows. In Section 2 we introduce the problem of safely planning and controlling the vehicle within an uncertain environment, while Section 3 outlines the Model Predictive Flexible trajectory Tracking Control (MPFTC) framework. In Section 4 we show how stability can be ensured when an infeasible reference trajectory is used, and in Section 5 we provide recursive feasibility guarantees for uncertain settings with suddenly appearing (pop-up) obstacles. Finally, in Section 6 we provide clarifying examples that illustrate the results derived in Sections 4 and 5.

1.1 Notation

We denote a discrete-time nonlinear system by

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

where $\mathbf{x}_k \in \mathbb{R}^{n_\mathbf{x}}$ and $\mathbf{u}_k \in \mathbb{R}^{n_\mathbf{u}}$ are the state and input vectors at time k, respectively. The state and inputs are subject to two categories of constraints: *a-priori known* constraints $h(\mathbf{x}, \mathbf{u}) : \mathbb{R}^{n_\mathbf{x}} \times \mathbb{R}^{n_\mathbf{u}} \rightarrow \mathbb{R}^{n_h}$; and *a-priori unknown* constraints $g(\mathbf{x}, \mathbf{u}) : \mathbb{R}^{n_\mathbf{x}} \times \mathbb{R}^{n_\mathbf{u}} \rightarrow \mathbb{R}^{n_g}$, i.e., the state and inputs must satisfy $h(\mathbf{x}, \mathbf{u}) \leq 0$ and $g(\mathbf{x}, \mathbf{u}) \leq 0$, where the inequalities are defined element-wise.

We use the notation $g_{n|k}(\mathbf{x}, \mathbf{u})$ to denote function g at time n, given the information available at time k. Moreover, we will denote by $g_n(\mathbf{x}, \mathbf{u}) := g_{n|\infty}(\mathbf{x}, \mathbf{u}) = g_{n|k}(\mathbf{x}, \mathbf{u})$, $\forall\, k \geq n$ the actual constraint, while in general $g_{n|k}(\mathbf{x}, \mathbf{u}) \neq g_n(\mathbf{x}, \mathbf{u})$, $\forall\, k < n$. Note that for a-priori known constraints $h_{n|k}(\mathbf{x}, \mathbf{u}) := h_n(\mathbf{x}, \mathbf{u})$ holds $\forall\, k$ by definition. We apply the same notation to state and inputs, e.g., $\mathbf{x}_{n|k}$ and $\mathbf{u}_{n|k}$ denote the predicted state and input at time n given the information available at the current time k. In addition, to denote a set of integers, we use $\mathbb{I}_a^b := \{a, a+1, ..., b\}$.

2 Problem Description

In this section we formulate the problem of *safely* planning and controlling the motion of a vehicle within an environment with static and moving obstacles, as a *Model Predictive Control Problem.*

The dynamical model of the vehicle, and the state and input constraints it is subject to, are denoted as in Section 1.1. Function h includes actuator limitations, design and safety (e.g., distance from the lane boundaries) constraints and is known beforehand. Function g models a-priori unknown constraints such as, e.g., pop-up (suddenly appearing) or moving obstacles, whose exact future motion trajectories are unknown. Our aim is then to control the vehicle motion described by (1) such that both known constraints $h_k(\mathbf{x}, \mathbf{u}) \leq 0$ and a-priori unknown constraints $g_k(\mathbf{x}, \mathbf{u}) \leq 0$ are satisfied at all times k.

Our first and essential objective is to guarantee safety of (1), which we define formally as follows.

Definition 1 (Safety)

A controller is said to be safe in a given set $\mathcal{S} \subseteq \mathbb{R}^{n_\mathbf{x}}$ if $\forall\, \mathbf{x} \in \mathcal{S}$ it generates control inputs $\mathbf{U} = \{\mathbf{u}_0, ..., \mathbf{u}_\infty\}$ and corresponding state trajectories $\mathbf{X} = \{\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_\infty\}$ such that $h_k(\mathbf{x}_k, \mathbf{u}_k) \leq 0$ and $g_k(\mathbf{x}_k, \mathbf{u}_k) \leq 0$, $\forall\, k \geq 0$.

Our second objective is to control the system such that the state and input $\mathbf{x}_k, \mathbf{u}_k$ track a parameterized reference trajectory $\mathbf{r}(\tau) := (\mathbf{r}^\mathbf{x}(\tau), \mathbf{r}^\mathbf{u}(\tau))$ *as closely as safety allows*. If the reference parameter τ is selected to be time, its natural dynamics are given by

$$\tau_{k+1} = \tau_k + t_\mathrm{s}, \tag{2}$$

where t_s is the sampling time for sampled-data systems and $t_\mathrm{s} = 1$ in the discrete-time framework. Given the presence of nonlinear dynamics and constraints, we frame the problem in the context of MPC. Note that if τ is forced to follow its natural dynamics (2), then the reference tracking problem in the absence of a-priori unknown constraints g is a standard MPC problem and, therefore, inherits all stability guarantees, but also a possibly aggressive behavior when the initial state is far from the reference. In order to tackle that issue and be able to deal with a-priori unknown constraints, we adopt next the concept of *Model Predictive Flexible Trajectory Tracking Control* introduced in [4].

3 Model Predictive Flexible Trajectory Tracking Control

The main idea in Model Predictive Flexible trajectory Tracking Control (MPFTC) is to avoid aggressive behaviors by adapting the dynamics of the reference trajectory by means of the parameter τ, which acts as a fictitious time for the reference, through relaxed dynamics given by

$$\tau_{k+1} = \tau_k + t_\mathrm{s} + v_k, \tag{3}$$

where v is an additional auxiliary control input and τ becomes an auxiliary state. Note that the system dynamics are unchanged and the fictitious time τ makes only the reference dynamics deviate from the natural ones.

We formulate the MPFTC problem as the following MPC problem

$$\begin{aligned}
V(\mathbf{x}_k, \tau_k) := \min_{\substack{\mathbf{x},\mathbf{u}\\ \tau, v}} \ & \sum_{n=k}^{k+N-1} q_\mathbf{r}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}, \tau_{n|k}) + w v_{n|k}^2 && \\
& + p_\mathbf{r}(\mathbf{x}_{k+N|k}, \tau_{k+N|k}) && \text{(4a)}\\
\text{s.t. } & \mathbf{x}_{k|k} = \mathbf{x}_k, \ \tau_{k|k} = \tau_k, && \text{(4b)}\\
& \mathbf{x}_{n+1|k} = f(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}), \qquad n \in \mathbb{I}_k^{k+N-1}, && \text{(4c)}\\
& \tau_{n+1|k} = \tau_{n|k} + t_\mathrm{s} + v_{n|k}, \qquad n \in \mathbb{I}_k^{k+N-1}, && \text{(4d)}\\
& h_n(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}) \leq 0, \qquad n \in \mathbb{I}_k^{k+N-1}, && \text{(4e)}\\
& g_{n|k}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}) \leq 0, \qquad n \in \mathbb{I}_k^{k+N-1}, && \text{(4f)}\\
& \mathbf{x}_{k+N|k} \in \mathcal{X}_\mathbf{r}^\mathrm{f}(\tau_{k+N|k}), && \text{(4g)}
\end{aligned}$$

where k is the current time, N is the prediction horizon, and $w > 0$ is the weight defining the cost associated with the auxiliary input $v_{n|k}$. In tracking MPC, typical choices for the stage and terminal costs are

$$q_{\mathbf{r}}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}, \tau_{n|k}) = \begin{bmatrix} \Delta\mathbf{x}_{n|k} \\ \Delta\mathbf{u}_{n|k} \end{bmatrix}^\top W \begin{bmatrix} \Delta\mathbf{x}_{n|k} \\ \Delta\mathbf{u}_{n|k} \end{bmatrix}, \tag{5}$$

$$p_{\mathbf{r}}(\mathbf{x}_{k+N|k}, \tau_{k+N|k}) = \Delta\mathbf{x}_{k+N|k}^\top P \Delta\mathbf{x}_{k+N|k}, \tag{6}$$

$$\Delta\mathbf{x}_{n|k} := \mathbf{x}_{n|k} - \mathbf{r}^{\mathbf{x}}(\tau_{n|k}), \quad \Delta\mathbf{u}_{n|k} := \mathbf{u}_{n|k} - \mathbf{r}^{\mathbf{u}}(\tau_{n|k}),$$

where the matrices $W \in \mathbb{R}^{(n_{\mathbf{x}}+n_{\mathbf{u}})\times(n_{\mathbf{x}}+n_{\mathbf{u}})}$ and $P \in \mathbb{R}^{n_{\mathbf{x}}\times n_{\mathbf{x}}}$ are symmetric positive-definite and $\mathbf{r}(\tau_{n|k}) = (\mathbf{r}^{\mathbf{x}}(\tau_{n|k}), \mathbf{r}^{\mathbf{u}}(\tau_{n|k}))$ is a user-provided reference trajectory. Note that the cost functions $q_{\mathbf{r}}$ and $p_{\mathbf{r}}$ depend on $\tau_{n|k}$ only through the reference trajectory and that the cost is built with convex quadratic forms just for simplicity, while the proposed framework can accommodate more general cost definitions. The predicted state and controls are defined as $\mathbf{x}_{n|k}$, $\tau_{n|k}$, and $\mathbf{u}_{n|k}$, $v_{n|k}$ respectively, and are subject to constraints (4b)-(4f). Constraint (4b) initializes the state prediction to the current system state $\mathbf{x}_k$, (4c)-(4d) impose that the predicted states generated from the system dynamics and the controls $\mathbf{u}_{n|k}$, (4e) enforces constraints stemming from, e.g., actuator physical limitations and reference trajectory bounds, while constraint (4f) forces the predicted state and controls to satisfy constraints imposed to avoid the collision with obstacles detected by a perception layer, hence, not known a-priori. Finally, (4g) is a terminal set constraint, which, differently from standard formulations, depends on the auxiliary state $\tau_{k+N|k}$ relative to the reference parameter. Note that, while the introduction of one additional state and control results in an increased computational complexity, such increase is typically small, since these variables have decoupled linear dynamics.

The stability proof for MPFTC has been provided in [4]. However, in Section 4 we will recall the necessary assumptions and state the stability theorem for completeness. We ought to stress that, in the absence of a-priori unknown constraints, the necessary assumptions for stability reduce to those commonly used to prove stability in standard MPC schemes. In the presence of a-priori unknown constraints, these assumptions may become too restrictive leading to the lack of recursive feasibility. Hence, in [4] we proposed to relax the standard assumptions while introducing new ones which can be summarized as follows. We first require one to "be able to predict a (possibly conservative) worst-case scenario" for the a-priori unknown constraints. Since the uncertainty typically becomes too large in rather short times, we need a second assumption which postulates the existence of a safe set, consisting of states for which the a-priori unknown constraints can be neglected. We will provide a more detailed discussion on this aspect in Section 5, where we will also discuss how to enforce these assumptions for autonomous driving.

4 Tracking an infeasible reference

We recall that our objective is to control the system (1), such that the state $\mathbf{x}_k$ tracks a *user-provided parameterized reference trajectory* $\mathbf{r}(\tau) = (\mathbf{r}^{\mathbf{x}}(\tau), \mathbf{r}^{\mathbf{u}}(\tau))$ as closely as possible. For the remainder of the chapter, we assume that the reference trajectory is parameterized with the time parameter t, and that it follows the natural dynamics (2). Furthermore, we will refer to any time dependence of the reference using the notation $(\mathbf{r}^{\mathbf{x}}_k, \mathbf{r}^{\mathbf{u}}_k) := (\mathbf{r}^{\mathbf{x}}(\tau_k), \mathbf{r}^{\mathbf{u}}(\tau_k))$, where τ is the fictitious time introduced in (3).

In order to prove stability, we recall the following standard assumptions, see, e.g., [26, 44].

Assumption 1 (System and cost regularity)

The system model f is continuous, and the stage cost $q_{\mathbf{r}} : \mathbb{R}^{n_{\mathbf{x}}} \times \mathbb{R}^{n_{\mathbf{u}}} \times \mathbb{R} \rightarrow \mathbb{R}_{\geq 0}$, and terminal cost $p_{\mathbf{r}} : \mathbb{R}^{n_{\mathbf{x}}} \times \mathbb{R} \rightarrow \mathbb{R}_{\geq 0}$, are continuous at the origin and satisfy $q_{\mathbf{r}}(\mathbf{r}^{\mathbf{x}}_k, \mathbf{r}^{\mathbf{u}}_k, \tau_k) = 0$, and $p_{\mathbf{r}}(\mathbf{r}^{\mathbf{x}}_k, \tau_k) = 0$. Additionally, $q_{\mathbf{r}}(\mathbf{x}_k, \mathbf{u}_k, \tau_k) \geq \alpha_1(\|\mathbf{x}_k - \mathbf{r}^{\mathbf{x}}_k\|)$ for all feasible $\mathbf{x}_k$, $\mathbf{u}_k$, and $p_{\mathbf{r}}(\mathbf{x}_k, \tau_k) \leq \alpha_2(\|\mathbf{x}_k - \mathbf{r}^{\mathbf{x}}_k\|)$, where α_1 and α_2 are $\mathcal{K}_\infty$-functions.

This assumption is common in MPC [26, 44] and can be relaxed in case one wants to account for so-called “economic” costs, see, e.g., [1, 19, 25, 39, 54, 55, 59, 53] for a generic theory and [31, 53] for applications to autonomous driving. The MPFTC framework can be extended to yield an approximate economic MPC, similar to the one proposed in [56, 57, 58, 16].

Assumptions 2, 3 are introduced as for standard MPC formulations, where no difference is made between a-priori known h and unknown constraints g. Nevertheless, we distinguish the case where the assumption is required to hold for h only from the case where the assumptions hold for both constraints, which is too restrictive in practice.

Assumption 2 (Reference feasibility)

The reference is feasible for the system dynamics, i.e., $\mathbf{r}^{\mathbf{x}}(t+t_{\mathrm{s}}) = f(\mathbf{r}^{\mathbf{x}}(t), \mathbf{r}^{\mathbf{u}}(t))$, and:

- ***a)*** *the reference satisfies the known constraints* (4e)*, i.e., $h_n(\mathbf{r}^{\mathbf{x}}(t_n), \mathbf{r}^{\mathbf{u}}(t_n)) \leq 0$, for all $n \in \mathbb{I}_0^\infty$;*
- ***b)*** *the reference satisfies the unknown constraints* (4f)*, i.e., $g_{n|k}(\mathbf{r}^{\mathbf{x}}(t_n), \mathbf{r}^{\mathbf{u}}(t_n)) \leq 0$, for all $n, k \in \mathbb{I}_0^\infty$.*

Assumption 2b is a strong assumption since it assumes that the reference is feasible for the unknown constraints for all future times, i.e., at time k the constraint $g_{n|k+1}$ is also assumed to be satisfied. This is clearly unrealistic for autonomous driving, since pedestrians and other vehicles may at some time cross the road and make the reference infeasible. Therefore, Assumption 2a serves as a relaxed version which is more realistic and will be used later, while dropping Assumption 2b.

Assumption 3 (Stabilizing Terminal Conditions)

There exists a parametric stabilizing terminal set $\mathcal{X}^{\mathrm{f}}_{\mathbf{r}}(t)$ and a terminal control law $\kappa^{\mathrm{f}}_{\mathbf{r}}(\mathbf{x}, t)$ yielding:

$$\mathbf{x}^{\kappa}_{+} = f(\mathbf{x}, \kappa^{\mathrm{f}}_{\mathbf{r}}(\mathbf{x}, t)), \qquad\qquad t_+ = t + t_{\mathrm{s}},$$

such that $p_{\mathbf{r}}(\mathbf{x}^{\kappa}_{+}, t_+) - p_{\mathbf{r}}(\mathbf{x}, t) \leq -q_{\mathbf{r}}(\mathbf{x}, \kappa^{\mathrm{f}}_{\mathbf{r}}(\mathbf{x}, t), t)$, and

a) $\mathbf{x} \in \mathcal{X}^{\mathrm{f}}_{\mathbf{r}}(t) \Rightarrow \mathbf{x}^{\kappa}_{+} \in \mathcal{X}^{\mathrm{f}}_{\mathbf{r}}(t_{+})$, *and* $h_n(\mathbf{x}, \kappa^{\mathrm{f}}_{\mathbf{r}}(\mathbf{x}, t)) \leq 0$, *for all* $n, k \in \mathbb{I}_0^{\infty}$*;*

b) $\mathbf{x} \in \mathcal{X}^{\mathrm{f}}_{\mathbf{r}}(t) \Rightarrow g_{n|k}(\mathbf{x}, \kappa^{\mathrm{f}}_{\mathbf{r}}(\mathbf{x}, t)) \leq 0$, *for all* $n, k \in \mathbb{I}_0^{\infty}$.

Similarly to Assumption 2b, Assumption 3b is also difficult to verify due to the unknown constraints. Hence, the milder Assumption 3a, which is standard in MPC settings, will be used later on whereas Assumption 3b will be dropped.

In order to track the reference $(\mathbf{r}^{\mathbf{x}}_k, \mathbf{r}^{\mathbf{u}}_k)$, we temporarily consider the following tracking MPC Problem

$$V(\mathbf{x}_k, \tau_k) := \min_{\mathbf{x},\mathbf{u}} \sum_{n=k}^{k+N-1} q_{\mathbf{r}}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}, \tau_n) + p_{\mathbf{r}}(\mathbf{x}_{k+N|k}, \tau_{k+N}) \tag{7a}$$

$$\text{s.t. } \mathbf{x}_{k|k} = \mathbf{x}_k, \tag{7b}$$

$$\mathbf{x}_{n+1|k} = f(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}), \qquad n \in \mathbb{I}_k^{k+N-1}, \tag{7c}$$

$$h(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}) \leq 0, \qquad n \in \mathbb{I}_k^{k+N-1}, \tag{7d}$$

$$\mathbf{x}_{k+N|k} \in \mathcal{X}^{\mathrm{f}}_{\mathbf{r}}(\tau_{k+N}). \tag{7e}$$

where, as opposed to Problem (4), τ follows the natural dynamics (2) instead of (3) and $g_{n|k}$ is removed, i.e., we consider a setting where there are no pop-up or other moving obstacles, or we make the restrictive assumption that these constraints cannot become active.

Assumptions 1-3 make it possible to derive the following standard (i.e., for feasible references) stability result.

Proposition 1 (Nominal Asymptotic Stability [3])

Suppose that the constraints $g_{n|k}$ *are inactive, the Assumptions 1, 2a, and 3a hold, and that the initial state* $(\mathbf{x}_k, \tau_k)$ *at time* k *belongs to the feasible set of Problem* (7)*. Then the system* (1) *in closed loop with the solution of* (7) *applied in receding horizon is an asymptotically stable system.*

Proof Since constraints $g_{n|k}$ are inactive by assumption, disregarding them in Problem (7) does not jeopardize feasibility. The rest of the proof follows from standard arguments, see, e.g., [9, 44]. □

Proposition 1 recalls the known stability results from the existing literature, which apply to tracking MPC schemes. We emphasize that, the design procedure resulting from Proposition 1 requires precomputing a feasible reference trajectory $(\mathbf{r}^{\mathbf{x}}_k, \mathbf{r}^{\mathbf{u}}_k)$ that satisfies Assumption 2. However, in practice, it may be convenient to use a reference trajectory that is infeasible w.r.t. the system dynamics, yet simpler to define. For example, in the design of a motion planner and controller for an autonomous vehicle driving on public roads, it would be convenient to just use as a reference trajectory an easily available lane centerline, which in general would not be feasible for kinematic or dynamic vehicle models.

While in standard MPC settings the stability with respect to an unreachable set point has been studied in [43], the approach therein applies to time-invariant infeasible references. In order to overcome such a limitation, we consider a setting where the reference can be time-varying and does not need to satisfy Assumption 2, and the terminal conditions (7e) do not need to hold at the reference trajectory, but in a neighborhood. To lay down the main result of this section (Theorem 2), we need to first introduce a few preliminary results.

Consider the optimal state and input trajectories obtained as the solution of the optimal control problem (OCP)

$$
\begin{aligned}
(\mathbf{x}^{\mathrm{r}}, \mathbf{u}^{\mathrm{r}}) := \lim_{M\to\infty} \arg\min_{\xi,\nu} \sum_{n=0}^{M-1} q_{\mathbf{r}}(\boldsymbol{\xi}_n, \boldsymbol{\nu}_n, \tau_n) + p_{\mathbf{r}}(\boldsymbol{\xi}_M, \tau_n) \qquad & \text{(8a)}\\
\text{s.t. } \boldsymbol{\xi}_0 = \mathbf{x}_0, \qquad & \text{(8b)}\\
\boldsymbol{\xi}_{n+1} = f(\boldsymbol{\xi}_n, \boldsymbol{\nu}_n), \qquad n \in \mathbb{I}_0^{M-1}, \qquad & \text{(8c)}\\
h(\boldsymbol{\xi}_n, \boldsymbol{\nu}_n) \le 0, \qquad n \in \mathbb{I}_0^{M-1}. \qquad & \text{(8d)}
\end{aligned}
$$

Note that, since constraints $g_{n|k}$ are considered to be inactive in this section, we do not include them in the OCP formulation in order to simplify the following analysis. Let $\mathbf{y}^{\mathrm{r}} := (\mathbf{x}^{\mathrm{r}}, \mathbf{u}^{\mathrm{r}})$ denote the solution of (8) and its optimal multipliers as $\boldsymbol{\lambda}^{\mathrm{r}}, \boldsymbol{\mu}^{\mathrm{r}}$. Hereafter, we will refer to the reference $\mathbf{y}^{\mathrm{r}}$ as the *feasible reference*, as it satisfies Assumption 2.

The result in Theorem 2 builds upon the stability theory for *economic MPC schemes*, where the cost is not of the *tracking type*. While the interested reader is referred to [1, 17, 21, 54] for an in-depth understanding of the stability analysis tools for economic MPC schemes, in this chapter we just recall that the main difference between economic and tracking MPC schemes is in the cost function, which satisfies

$$
q_{\mathbf{r}}(\mathbf{x}_k^{\mathrm{r}}, \mathbf{u}_k^{\mathrm{r}}, \tau_k) = 0,\ q_{\mathbf{r}}(\mathbf{x}_k, \mathbf{u}_k, \tau_k) > 0,\ \forall\, \mathbf{x}_k \neq \mathbf{x}_k^{\mathrm{r}},\ \mathbf{u}_k \neq \mathbf{u}_k^{\mathrm{r}}, \tag{9}
$$

in tracking schemes but not in economic ones. While the MPC scheme built on a reference trajectory satisfying Assumption 2 is of a tracking type, an infeasible reference trajectory yields an economic scheme. Hence, in order to retrieve a tracking cost from the economic one, we introduce the following *rotated costs*

$$
\begin{aligned}
\bar{q}_{\mathbf{r}}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}, \tau_n) &:= q_{\mathbf{r}}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}, \tau_n) - q_{\mathbf{r}}(\mathbf{x}_n^{\mathrm{r}}, \mathbf{u}_n^{\mathrm{r}}, \tau_n) \\
&\quad + \boldsymbol{\lambda}_n^{\mathrm{r}\top}(\mathbf{x}_{n|k} - \mathbf{x}_n^{\mathrm{r}}) - \boldsymbol{\lambda}_{n+1}^{\mathrm{r}\top}(f_n(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}) - f_n(\mathbf{x}_n^{\mathrm{r}}, \mathbf{u}_n^{\mathrm{r}})),
\end{aligned} \tag{10}
$$

$$
\bar{p}_{\mathbf{r}}(\mathbf{x}_{n|k}, \tau_n) := p_{\mathbf{r}}(\mathbf{x}_{n|k}, \tau_n) - p_{\mathbf{r}}(\mathbf{x}_n^{\mathrm{r}}, \tau_n) + \boldsymbol{\lambda}_n^{\mathrm{r}\top}(\mathbf{x}_{n|k} - \mathbf{x}_n^{\mathrm{r}}), \tag{11}
$$

which are commonly used in economic MPC. The rotated cost essentially shifts the stage and terminal costs $q_{\mathbf{r}}$ and $p_{\mathbf{r}}$, so that the minimum is attained at the *feasible reference* $(\mathbf{x}^{\mathrm{r}}, \mathbf{u}^{\mathrm{r}})$, i.e., $\bar{q}_{\mathbf{r}}(\mathbf{x}_k^{\mathrm{r}}, \mathbf{u}_k^{\mathrm{r}}, \tau_k) = 0$, $\bar{p}_{\mathbf{r}}(\mathbf{x}_k^{\mathrm{r}}, \tau_k) = 0$. However, in order to ensure that the rotated costs remain positive definite, we must assume that the system dynamics are linear time-varying, i.e., that Assumption 4 (which we introduce next) must hold.

The issue of tracking an infeasible time-varying reference has been studied in [3], under the additional assumption of linear time-varying (LTV) system dynamics. This assumption is technical and in practice we expect that the results can be extended to the fully nonlinear case, which will be the subject of future research.

Assumption 4 *The system dynamics f are linear time-varying, i.e.,*

$$\mathbf{x}_{k+1} = f_k(\mathbf{x}_k, \mathbf{u}_k) = A_k \mathbf{x}_k + B_k \mathbf{u}_k. \tag{12}$$

In order to construct a problem that tracks the *feasible reference* obtained from (8), we formulate the following *ideal* formulation

$$V^{\mathrm{i}}(\mathbf{x}_k, \tau_k) = \min_{\mathbf{x},\mathbf{u}} \sum_{n=k}^{k+N-1} q_{\mathbf{r}}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}, \tau_n) + p_{\tilde{\mathbf{y}}^{\mathrm{r}}}(\mathbf{x}_{k+N|k}, \tau_{k+N}) \tag{13a}$$

$$\text{s.t.} \;\; \text{(4b)} - \text{(4e)}, \;\; \mathbf{x}_{k+N|k} \in \mathcal{X}^{\mathrm{f}}_{\mathbf{y}^{\mathrm{r}}}(\tau_{k+N}), \tag{13b}$$

where

$$\tilde{\mathbf{y}}^{\mathrm{r}}_k := \arg\min_{\mathbf{x}} p_{\mathbf{y}^{\mathrm{r}}}(\mathbf{x}, \tau_k) - \lambda^{\mathrm{r}\top}_k(\mathbf{x} - \mathbf{x}^{\mathrm{r}}_k). \tag{14}$$

We refer to this formulation as being *ideal* since the terminal conditions are in general not known, unless one solves OCP (8).

Theorem 1 *Suppose that*

1. *Assumption 1 holds,*
2. *Problem* (8) *is feasible,*
3. *Assumption 3 holds for $\bar{q}_{\mathbf{r}}$ and $\bar{p}_{\tilde{\mathbf{y}}^{\mathrm{r}}}$, with terminal set $\mathcal{X}_{\mathbf{y}^{\mathrm{r}}}$,*
4. *Assumption 4 holds for the system dynamics.*

Then, the system (1) *in closed-loop with the ideal MPC* (13) *is asymptotically stabilized to the optimal trajectory* $\mathbf{x}^{\mathrm{r}}$. □

Theorem 1 establishes that an MPC problem can be formulated using an *infeasible reference*, which stabilizes system (1) to the *feasible reference* obtained from Problem (8), provided that appropriate terminal conditions are used. The remaining issue, however, is to express the terminal constraint set as a positive invariant set containing $\mathbf{x}^{\mathrm{r}}$, and a terminal control law that stabilizes the system to $\mathbf{x}^{\mathrm{r}}$. To that end, one needs to have prior knowledge of the *feasible reference*, i.e., Problem (8) needs to be solved. Instead, we consider expressing terminal conditions that are based on an *approximately feasible* reference. In that case, asymptotic stability in the sense of Proposition 1 cannot be proven. We will therefore resort to input-to-state stability for the closed-loop system, where the input will be a terminal reference $\mathbf{y}^{\mathrm{f}}$ satisfying the following assumption.

Assumption 5 (Approximate feasibility of the reference)

The reference $\mathbf{y}^{\mathrm{f}}$ *satisfies the constraints* (4e)*, i.e., $h(\mathbf{x}^{\mathrm{f}}_n, \mathbf{u}^{\mathrm{f}}_n) \leq 0$, $n \in \mathbb{I}_k^{k+N-1}$, for all $k \in \mathcal{N}^+$. Additionally, recursive feasibility holds for both Problem* (7) *and* (13) *when the system is controlled in closed-loop using the feedback from Problem* (7).

Assumption 5 sets a rather mild requirement from a practical standpoint. Using an infeasible reference for simplicity, or approximating system dynamics to capture the most relevant dynamics of the system ($\|\mathbf{x}^{\mathrm{f}}_{n+1} - f_n(\mathbf{x}^{\mathrm{f}}_n, \mathbf{u}^{\mathrm{f}}_n)\| \leq \epsilon$, for some small ϵ) is not uncommon in practice. In particular, in a practical setting we can select $\mathbf{y}^{\mathrm{f}} = \mathbf{r}(t_{k+N})$, or in an ideal setting $\mathbf{y}^{\mathrm{f}} = \mathbf{y}^{\mathbf{r}}(t_{k+N})$. To that end, we define the following closed-loop dynamics

$$\mathbf{x}_{k+1}(\mathbf{y}^{\mathrm{f}}) = f_k(\mathbf{x}_k, \mathbf{u}_{\mathrm{MPC}}(\mathbf{x}_k, \mathbf{y}^{\mathrm{f}})) = \bar{f}_k(\mathbf{x}_k, \mathbf{y}^{\mathrm{f}}), \tag{15}$$

where $\mathbf{u}_{\mathrm{MPC}}$ is obtained as $\mathbf{u}^{\star}_{k|k}$ solving Problem (7) in case $\mathbf{y}^{\mathrm{f}} = \mathbf{r}$; and as $\mathbf{u}^{\mathrm{i}}_{k|k}$ solving the *ideal* Problem (13) in case $\mathbf{y}^{\mathrm{f}} = \mathbf{y}^{\mathrm{r}}$.

We are now ready to state the main results of this section.

Theorem 2 *Suppose that*

1. *Problem* (8) *is feasible,*
2. *Assumptions 1 and 3 hold for the reference* $\mathbf{y}^{\mathrm{r}}$ *with costs* $\bar{q}_{\mathbf{r}}$ *and* $\bar{p}_{\mathbf{y}^{\mathrm{r}}}$ *and terminal set* $\mathcal{X}_{\mathbf{y}^{\mathrm{r}}}$*,*
3. *Problem* (7) *and Problem* (13) *are feasible at time* k *with initial state* $(\mathbf{x}_k, t_k)$*,*
4. *the reference* $\mathbf{y}^{\mathrm{f}}$*, with terminal set* $\mathcal{X}^{\mathrm{f}}_{\mathbf{y}^{\mathrm{f}}}$*, satisfies Assumption 5.*

Then, system (15) *obtained from* (1) *in closed-loop with MPC formulation* (7) *is ISS.*

□

This theorem proves that if an infeasible reference is used, system (1) does not converge exactly to the (unknown) optimal trajectory from OCP (8), but to a neighborhood around it which depends on how inaccurate the terminal reference is. We note, however, that the effect of the terminal condition on the closed-loop trajectory decreases as the prediction horizon N increases [21, 54].

This section has so far considered the a-priori unknown constraint $g_{n|k}$ to be inactive, i.e., no road users or other obstacles are present, in order to simplify the analysis. In the next section, we consider settings where $g_{n|k}$ may not be ignored, e.g., one has to ensure collision-avoidance w.r.t other road users.

5 Safety-Enforcing MPC

The aim of this section is to tackle the issues posed by the presence of the a-priori unknown constraints (4f), which in Section 4 have been assumed to always remain inactive. While we cast the problem in the framework of MPFTC, we stress that the developments proposed to enforce safety are independent of the specific tracking scenario, i.e., flexible trajectory, path, setpoint, etc., and can also be deployed in the context of Model Predictive Path Following Control (MPFC) proposed in [18, 20].

We introduce the following assumption, imposing some structure on g that is needed in order to ensure that the feasibility of a solution is preserved between consecutive time instances.

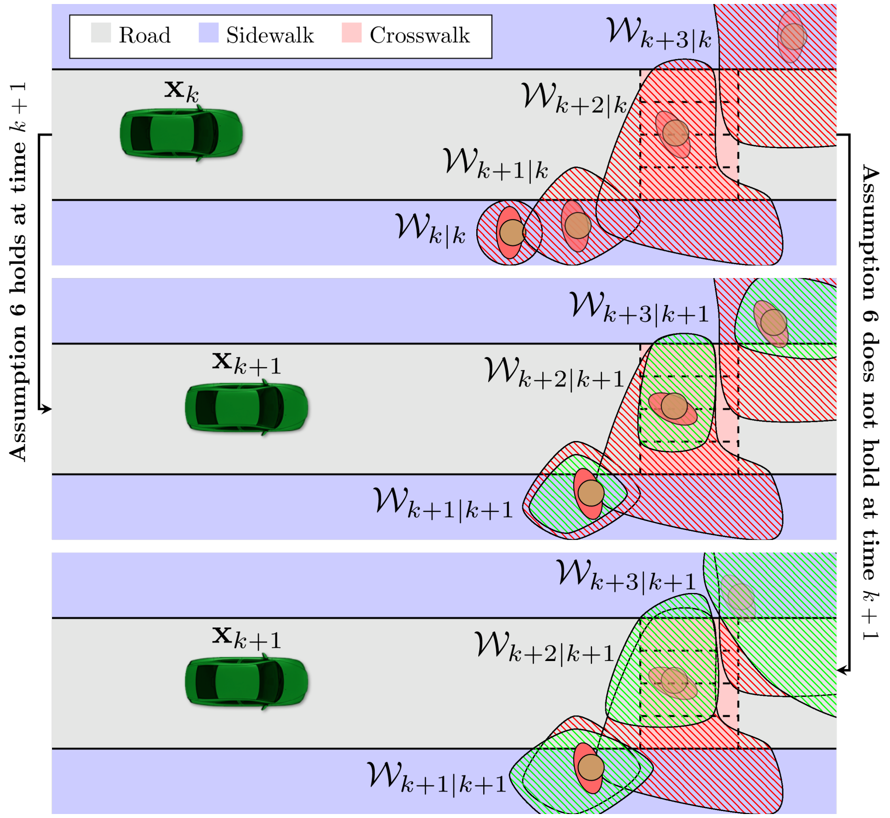

Fig. 1 The top panel shows an initial prediction for a pedestrian at time k, while the middle and bottom panels show different predictions at time $k+1$. The middle panel illustrates a model that satisfies (18), i.e., satisfying Assumption 6, while the bottom panel does not satisfy (18), hence, Assumption 6 is also not satisfied.

Assumption 6 (Unknown constraint dynamics)

The a-priori unknown constraints satisfy $g_{n|k+1}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}) \leq g_{n|k}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k})$, *for all* $n \geq k$.

While Fig. 1 gives a visual interpretation of the requirement in Assumption 6, at this stage it is natural to wonder how function g should be constructed such that Assumption 6 is satisfied. In order to answer this question, we will first provide a formal description of how g can be constructed, and then provide some examples which are relevant to autonomous driving to provide more intuition on the meaning of Assumption 6.

We introduce the function $\gamma(\mathbf{x}, \mathbf{u}, \mathbf{w}) : \mathbb{R}^{n_\mathbf{x}} \times \mathbb{R}^{n_\mathbf{u}} \times \mathbb{R}^{n_\mathbf{w}} \to \mathbb{R}^{n_g}$ and the uncertain variable $\mathbf{w}_{n|k} \in \mathcal{W}_{n|k} \subseteq \mathbb{R}^{n_\mathbf{w}}$ whose bounded support is a subset of set $\mathcal{W}_{n|k}$, lumping all the uncertainty related to the a-priori unknown constraints. Note that we only require knowledge on a superset of the support of $\mathbf{w}_{n|k}$, which can in principle even be deterministic, in which case its probability distribution is a Dirac and the superset $\mathcal{W}_{n|k}$ can be any set of measure 1.

Then we define

$$g_{n|k}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}) := \max_{\mathbf{w}_{n|k} \in \mathcal{W}_{n|k}} \gamma_{n|k}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}, \mathbf{w}_{n|k}). \tag{16}$$

This formulation implies robust constraint satisfaction, i.e.,

$$g_{n|k}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}) \leq 0 \qquad \Leftrightarrow \qquad \begin{cases} \gamma_{n|k}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}, \mathbf{w}_{n|k}) \leq 0, \\ \forall\ \mathbf{w}_{n|k} \in \mathcal{W}_{n|k}. \end{cases}$$

In a general setting, $\mathbf{w}_{n|k}$ is the state of the dynamical system

$$\mathbf{w}_{n+1|k} = \omega(\mathbf{w}_{n|k}, \xi_{n|k}, \mathbf{x}_{n|k}, \mathbf{u}_{n|k}), \tag{17}$$

with associated control variable $\xi_{n|k} \in \Xi \subseteq \mathbb{R}^{m_\xi}$, acting as a source of (bounded) noise. The function ω describes the dynamics, and the explicit dependence on $\mathbf{x}_{n|k}$, $\mathbf{u}_{n|k}$ models possible interactions between the uncertainty and system (1). This can be the case, for example, of a pedestrian, a bicycle or a human-driven car which interacts with the vehicle whose motion needs to be planned and controlled. With model (17) reachability analysis tools can be used to predict the future evolution of the outer-approximations of the sets $\mathcal{W}_{n|k}$.

$$\mathcal{W}_{n+1|k}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}) :\supseteq \{\, \omega(\mathbf{w}_{n|k}, \xi_n, \mathbf{x}_{n|k}, \mathbf{u}_{n|k}) \mid \mathbf{w}_{n|k} \in \mathcal{W}_{n|k},\ \forall \xi_n \in \Xi \,\}, \tag{18}$$

for some initial $\mathcal{W}_{k|k} = \mathbf{w}_{k|k}$.

The introduction of the uncertainty sets (18) allows one to model road users which (a) are detectable by the sensors, and (b) are either beyond the onboard sensors range or hidden by other obstacles, as depicted in Fig. 2. For type (a), the model (16), (17) does not underestimate the set of future states that can be reached by the road users. For type (b), the uncertainty model must predict the possibility that a road user appears at any time either at the boundary of the sensor range or from behind an obstacle.

We can now state the following result.

Lemma 1 *Suppose that $g_{n|k}$ is defined according to* (16) *with $\mathcal{W}_{n|k}$ satisfying* (18)*. Then, Assumption 6 holds.*

Note that this lemma amounts to assuming that the uncertainty in $g_{n|k}$ cannot increase as additional information becomes available (from either the onboard sensors or through communication links). Furthermore, a direct consequence of Assumption 6 is $g_{n|k}(\mathbf{r}^\mathbf{x}(t_n), \mathbf{r}^\mathbf{u}(t_n)) \leq 0 \implies g_{n|k+1}(\mathbf{r}^\mathbf{x}(t_n), \mathbf{r}^\mathbf{u}(t_n)) \leq 0$. We provide the following clarifying illustrations in order to better understand Lemma 1 and Assumption 6.

Fig. 1 shows the difference between having a model that satisfies Lemma 1, and one that does not. The middle panel shows that the predictions made at time $k+1$ belong to a subset of the previous predictions made at time k, i.e., $\mathcal{W}_{n|k+1} \subseteq \mathcal{W}_{n|k}$,

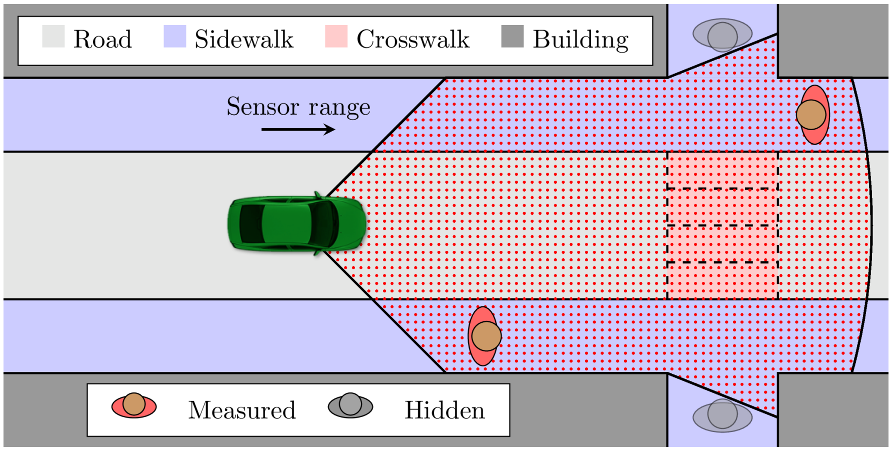

Fig. 2 Due to limited sensing capabilities, it is not possible to directly measure all road users in the environment. Therefore, one must assume that hidden road users may appear outside the sensor range at all times.

which satisfies Lemma 1. The bottom panel on the other hand illustrates predictions made at time $k + 1$, where Lemma 1 does not hold. Fig. 2 illustrates the fact that road users in the environment may be occluded due to limited sensing capabilities. In that case, in order to satisfy Assumption 6 at all times, one needs to model the possibility that a road user might appear at the boundary of the sensor range.

We recall that, in the context of autonomous driving, the constraints $g_{n|k}$ could enforce avoiding collisions with obstacles (e.g., other road users) detected by the sensors, whose behavior can just be predicted to some limited extent. Hence, Assumption 6 amounts to assuming that the uncertainty on, e.g., position and velocity estimates of the detected objects at a specific time instance cannot increase as additional information becomes available. We note however that limited sensor range makes it impossible to detect obstacles which are too far away. To ensure satisfaction of Assumption 6 one can adopt a worst-case approach which ensures that the predicted trajectory $\mathbf{x}_{n|k}$ may never leave the sensor range, and by also assuming that new obstacles appear at the boundary of the sensor range at all times. A visual example of this shown in Fig. 3, where the planned trajectory is forced to remain within the sensor range at all times.

Enforcing *safety* of the controller (4) according to Definition 1 requires the controller to be *recursively feasible*. Hence, the terminal constraint (4g) needs to be designed in order to guarantee its satisfaction despite of the presence of the constraints (4f), which can grow unbounded in time, such that no state is safe, see Fig. 4. We therefore design the terminal conditions by assuming the existence of a *safe set*, where the constraints (4f) are guaranteed to be satisfied *regardless*. This will allow us to rely on standard approaches in MPC [9, 34, 51] which are based on the existence of a robust invariant set.

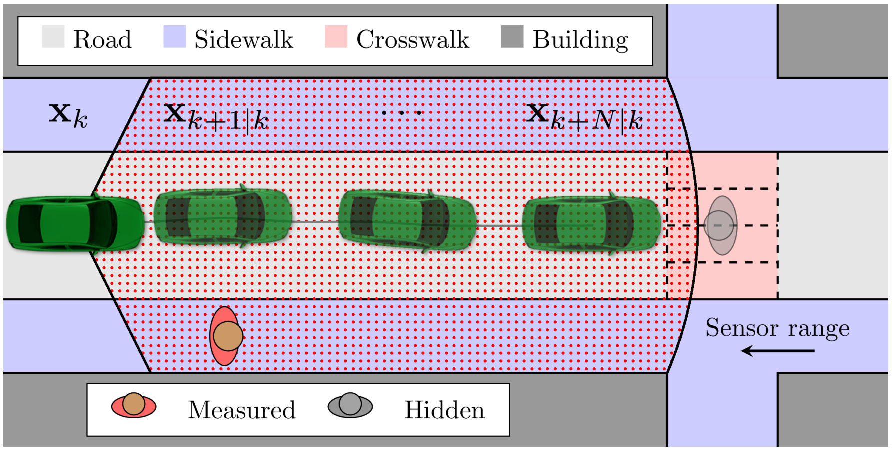

Fig. 3 Since it is impossible to know what lies ahead of the sensing range, the planned trajectory $\{\mathbf{x}_{k|k}, \mathbf{x}_{k+1|k}, \ldots, \mathbf{x}_{k+N|k}\}$ must be forced to remain within the sensor range.

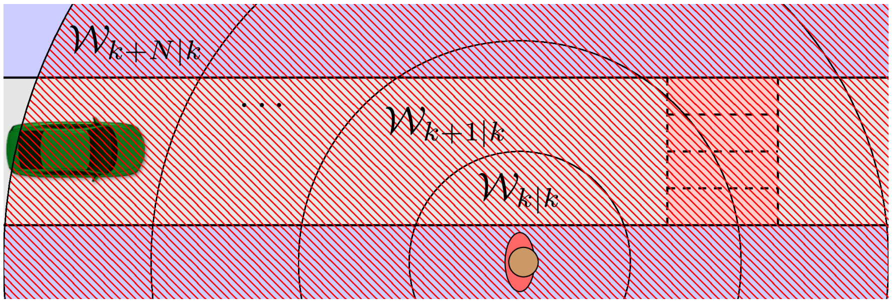

Fig. 4 As the uncertainty of the pedestrian predictions grow in time, the collision-free region of the vehicle drastically shrinks. Hence, after some time, any model will predict that a pedestrian can be anywhere, such that no state is safe.

Assumption 7 *There exists a robust invariant set denoted $\mathcal{X}_{\text{safe}}(\tau_{n|k}) \subseteq \mathbb{R}^{n_x}$ such that for all $\mathbf{x}_{n|k} \in \mathcal{X}_{\text{safe}}(\tau_{n|k})$ there exists a safe control set $\mathcal{U}_{\text{safe}}(\mathbf{x}_{n|k}, \tau_{n|k}) \subseteq \mathbb{R}^{n_u+1}$ entailing that $f(\mathbf{x}_{n|k}, \mathbf{u}_{\text{safe}}) \in \mathcal{X}_{\text{safe}}(\tau_{n|k} + t_{\text{s}} + v_{\text{safe}})$, and $h_n(\mathbf{x}_{n|k}, \mathbf{u}_{\text{safe}}) \leq 0$, for all $(\mathbf{u}_{\text{safe}},\ v_{\text{safe}}) \in \mathcal{U}_{\text{safe}}(\mathbf{x}_{n|k}, \tau_{n|k})$ and for all $n \geq k$. Moreover, for all $\mathbf{x}_{n|k} \in \mathcal{X}_{\text{safe}}(\tau_{n|k})$ the a-priori unknown constraints can never be violated, i.e., by construction $g_{n|k}(\mathbf{x}_{n|k}, \mathbf{u}_{\text{safe}}) \leq 0$ for all $\mathbf{x}_{n|k} \in \mathcal{X}_{\text{safe}}(\tau_{n|k})$ and $(\mathbf{u}_{\text{safe}},\ v_{\text{safe}}) \in \mathcal{U}_{\text{safe}}(\mathbf{x}_{n|k}, \tau_{n|k})$.*

While this assumption might seem strong, it only postulates the existence of *known* safe configurations for system (1). However, if no such configurations exist, then the controller based on Problem (4) is intrinsically unsafe. On the other hand, if such configurations do exist for (1), then the safe set $\mathcal{X}_{\text{safe}}$ is non-empty and invariant. Note that the safe configuration depends on system (1), the problem setting, and must be known a-priori.

Example 1 *Many practical settings where safety is emphasized consider a system to be safe at steady state, in which case the safety set $\mathcal{X}_{\text{safe}}$ can be formulated as*

$$\mathcal{X}_{\text{safe}}(\tau_k) := \{\, \mathbf{x} \,|\, \mathbf{x} = f(\mathbf{x}, \mathbf{u}),\ h_k(\mathbf{x}, \mathbf{u}) \leq 0,\ m_k(\mathbf{x}, \mathbf{u}) \leq 0 \,\}, \tag{19}$$

where function m_k defines additional constraints which might be needed in the set definition. A notable example for automotive settings is that a vehicle parked in a safe configuration, e.g., a parking lot, emergency lane or any other safe environment that can be modeled by m_k, is not responsible for collisions with other road users. This reasoning can be applied to the setting illustrated in Fig. 4, where the uncertainty grows such that the only safe thing to do is to force the vehicle to a complete stop.

The introduction of Assumption 7, allows us to drop Assumption 3b, so that we can build our approach based on standard strategies in MPC [9, 34, 51], i.e., we rely on stabilizing terminal control laws $\kappa^{\text{s}}_{\mathbf{r}}(\mathbf{x}, t)$ and sets $\mathcal{X}^{\text{s}}_{\mathbf{r}}(t)$ satisfying Assumption 3a. In order to obtain recursive feasibility also with respect to a-priori unknown constraints, we rely on the safe set $\mathcal{X}_{\text{safe}}$ to introduce the following terminal set

$$\begin{aligned}
\mathcal{X}^{\text{f}}_{\mathbf{r}}(\tau_{k+N|k}) := \{\mathbf{x}_{k+N|k} \mid \exists\, &\mathbf{u}_{n|k}, v_{n|k}, && \text{(20a)}\\
&\tau_{n+1|k} = \tau_{n|k} + t_{\text{s}} + v_{n|k}, && \text{(20b)}\\
&\mathbf{x}_{n+1|k} = f(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}), && \text{(20c)}\\
&h_n(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}) \leq 0, && \text{(20d)}\\
&g_{n|k}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}) \leq 0, && \text{(20e)}\\
&\mathbf{x}_{n|k} \in \mathcal{X}^{\text{s}}_{\mathbf{r}}(\tau_{n|k}), && \text{(20f)}\\
&\mathbf{x}_{k+M|k} \in \mathcal{X}_{\text{safe}}(\tau_{k+M|k}) \subseteq \mathcal{X}^{\text{s}}_{\mathbf{r}}(\tau_{k+M|k}), && \text{(20g)}\\
&\text{(20a)} - \text{(20f)},\ \forall n \in \mathbb{I}_{k+N}^{k+M-1}\}, && \text{(20h)}
\end{aligned}$$

where $M \geq N$ is a degree of freedom. Note that the construction of (20) implies that $\mathcal{X}_{\text{safe}}(\tau) \subseteq \mathcal{X}^{\text{s}}_{r}(\tau), \forall \tau \geq 0$. If (20) is a non-empty set we are guaranteed that for all $\mathbf{x} \in \mathcal{X}^{\text{f}}_{\mathbf{r}}(\tau)$ a terminal control law exists, which steers the states to the safe set.

In order to provide a practical approach to design the terminal control law, we propose to first design a control law $\kappa^{\text{s}}_{\text{r}}$ as one would do in standard MPC formulations, i.e., by ignoring a-priori unknown constraints g and by forcing the time in the reference to evolve according to its true dynamics. We can then define the terminal control law $(\kappa^{\text{f}}_{\mathbf{r}}(\mathbf{x}_{k+N|k}, \tau_{k+N|k}), v^{\text{f}}_{\mathbf{r}}(\mathbf{x}_{k+N|k}, \tau_{k+N|k}))$ by using $\kappa^{\text{s}}_{\text{r}}$, as the solution of

$$\begin{aligned}
\min_{\mathbf{u}, v}\ \ & \|\mathbf{u} - \kappa^{\text{s}}_{\text{r}}(\mathbf{x}_{k+N|k}, \tau_{k+N|k})\|_2 + v^2 && \text{(21a)}\\
\text{s.t.}\ \ & f(\mathbf{x}_{k+N|k}, \mathbf{u}) \in \mathcal{X}^{\text{f}}_{\mathbf{r}}(\tau_{k+N|k} + t_{\text{s}} + v), && \text{(21b)}\\
& h_{k+N}(\mathbf{x}_{k+N|k}, \mathbf{u}) \leq 0, && \text{(21c)}\\
& g_{k+N|k}(\mathbf{x}_{k+N|k}, \mathbf{u}) \leq 0. && \text{(21d)}
\end{aligned}$$

The idea behind the terminal set (20) is to ensure safety by forcing the system to be able to reach a safe set $\mathcal{X}_{\text{safe}}$ in a finite amount of time $M - N \geq 0$, while always remaining inside a stabilizing set $\mathcal{X}_{\text{r}}^{\text{s}}(t)$ around the reference. Note that M is a parameter which can be used to tune the stabilizing terminal safe set and, consequently, the NMPC scheme (4). If $M = N$, then the terminal set coincides with the safe set, possibly limiting the capabilities of the terminal control law, i.e., $\kappa_{\mathbf{r}}^{\text{f}}(\mathbf{x},\tau) \neq \kappa_{\mathbf{r}}^{\text{s}}(\mathbf{x},\tau)$ and $\nu_{\mathbf{r}}^{\text{f}}(\mathbf{x},\tau) \neq 0$. On the other hand, if $M \gg N$, the computational complexity of $\mathcal{X}_{\mathbf{r}}^{\text{f}}$ can become excessive.

Theorem 3 (Recursive Feasibility)

Suppose that Assumptions 1, 2a, 3a, 6, and 7 hold, and that Problem (4) *is feasible for the initial state* $(\mathbf{x}_k, \tau_k)$*, with terminal set and terminal controllers given by* (20) *and* (21)*, respectively. Then, system* (1)-(3) *in closed loop with the solution of* (4) *applied in receding horizon is safe (recursively feasible) at all times.*

While Theorem 3 only proves recursive feasibility, the presence of obstacles makes it more difficult to discuss closed-loop stability. We note, however, that if the a-priori unknown constraints become inactive, then the proposed formulation yields nominal asymptotic stability.

Next we consider two simulation examples to illustrate the theory from Sections 4 and 5.

6 Simulations

In this section we present two simulations related to autonomous driving settings in order to illustrate the theory from Sections 4 and 5. In Section 6.1 we first show how the closed-loop behavior is affected when an infeasible reference is used. Then, in Section 6.2 we introduce moving and pop-up obstacles to show how recursive feasibility is ensured through Theorem 3.

For both simulations we consider the single-track vehicle model with kinematics

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{\delta} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} v\cos(\psi) \\ v\sin(\psi) \\ \frac{v}{l}\tan(\delta) \\ \omega \\ a \end{bmatrix}, \tag{22}$$

where x, y are the position coordinates in a global frame, v is the velocity, ψ is the orientation angle, l is the wheelbase length, δ is the steering wheel angle, and a and ω denote the acceleration and steering wheel angle rate, respectively. Since one of the objectives is to track a user-defined reference $\mathbf{r}$, it is possible to geometrically derive the following vehicle kinematics in the frame of the reference path [37]

$$\begin{bmatrix} \dot{s} \\ \dot{e}_y \\ \dot{e}_\psi \\ \delta \\ \dot{v} \end{bmatrix} = \begin{bmatrix} v\cos(e_\psi)(1-\kappa^{\mathrm{r}}(s)e_y)^{-1} \\ v\sin(e_\psi) \\ vl^{-1}(\tan(\delta)-\tan(\delta^{\mathrm{r}}(s))) \\ \omega \\ a \end{bmatrix}, \; \mathbf{x} = \begin{bmatrix} e_y \\ e_\psi \\ \delta \\ v \end{bmatrix}, \; \mathbf{u} = \begin{bmatrix} a \\ \omega \end{bmatrix} \tag{23}$$

where s is the longitudinal position along the path, κ^{r} is the path curvature, e_y is the lateral displacement error, e_ψ is the yaw error with respect to the reference $\mathbf{r}$ and δ^{r} is the reference steering angle. Note that we consider s to be an auxiliary state since we are only interested in tracking the velocity and not the longitudinal position. For both simulations, we assume that system (23) is subject to the following known constraints

$$\begin{gathered} \|e_y\| \le 0.4, \; \|e_\psi\| \le 0.16, \; \|\delta\| \le 0.53, \\ 0 \le v \le 50/3.6, \; -5 \le a \le 3, \; \|\omega\| \le 0.35, \end{gathered}$$

while the stage cost matrices in (5) are

$$W = \mathrm{blockdiag}(Q, R), \; Q = \mathrm{diag}(1, 10, 1, 1), \; R = \mathrm{diag}(1, 1). \tag{24}$$

In order to compute the stabilizing terminal set $\mathcal{X}_{\mathrm{s}}^{\mathrm{f}}(t)$, we decouple the longitudinal and lateral kinematics. Using an an LQR controller with costs $Q^{\mathrm{lon}} = 1$, $R^{\mathrm{lon}} = 50$ one can then obtain the feedback gain $K = 0.14$ a terminal cost $P^{\mathrm{long}} = 72.63$ with corresponding terminal set

$$\mathcal{X}_s^{\mathrm{lon}}(\tau) := \{v \mid -5 \le \; -K(v - \mathbf{r}^{\mathbf{x}_v}(\tau)) \le 3 \,\}. \tag{25}$$

For the terminal set of the lateral kinematics, we consider the velocity to be an uncertain parameter and linearize the system (23) on the reference to obtain the Linear Parameter Varying (LPV) system $(A(v), B(v)) \in \mathbb{R}^{3\times 3} \times \mathbb{R}^3$. Then, by considering a nominal velocity $v^{\mathrm{nom}} = 28/3.6$ m/s, we use the feedback gain K, obtained from an LQR controller with tuning $Q^{\mathrm{lat}} = \mathrm{blockdiag}(1,\ 100,\ 2)$ and $R^{\mathrm{blockdiag}} = 1$, to stabilize the LPV system by considering the following polytopic system

$$\Gamma := \{(A, B) \in \mathbb{R}^{3\times 3} \times \mathbb{R}^3 : A = A(v), B = B(v), v \in [1, 50/3.6] \,\}, \tag{26}$$

for velocities $v \in [1, 50/3.6]$ m/s. We then use the MPT toolbox [30] to compute the terminal set [8] for the polytopic system (26) and obtain that

$$X_{\mathbf{r}}^{\mathrm{lat}}(\tau) := \{H([e_y, e^{\psi}, \delta]^\top - [0, 0, \delta^{\mathrm{ref}}(\tau)]^\top) \le b\}. \tag{27}$$

We note that, while we compute the terminal set for the polytopic system (26), one can also use the methods presented in [27, 28] to compute low-complexity invariant sets for linear parameter-varying models. By solving the linear matrix inequalities that satisfy the Lyapunov equations for the polytopic system, we also obtain the terminal cost

$$P^{\mathrm{lat}} = \begin{bmatrix} 402.37 & 839.90 & 323.00 \\ 839.90 & 7272.88 & 2781.57 \\ 323.00 & 2781.57 & 2556.74 \end{bmatrix} \tag{28}$$

Finally, the terminal set $\mathcal{X}_{\mathrm{s}}^{\mathrm{f}}(\tau)$ can then be constructed as

$$\mathcal{X}_{\mathrm{s}}^{\mathrm{f}}(\tau) := \{\mathbf{x} \mid [e_y, e_\psi, \delta] \in \mathcal{X}_{\mathbf{r}}^{\mathrm{lat}}(\tau),\ v \in \mathcal{X}_{\mathrm{r}}^{\mathrm{long}}(\tau)\}, \tag{29}$$

with terminal cost $P = \mathrm{blockdiag}(P^{\mathrm{lat}}, P^{\mathrm{lon}})$.

The following simulations were implemented in Matlab and interfaced with Acados [49] and CasADi [2], together with solvers IPOPT [50] and HPIPM [22].

6.1 ISS: Stability with Infeasible Reference

In this section we show the closed-loop behavior of Problem (7) when using an infeasible reference. To that end, we consider a reference with a discontinuous curvature, i.e., the steering angle reference is discontinuous,

$$\mathbf{r}^{\mathbf{x}}(\tau) = [0,\ 0,\ \delta^{\mathrm{r}}(\tau),\ 50/3.6]^\top, \mathbf{r}^{\mathbf{u}}(\tau) = [0,\ 0]^\top, \tag{30}$$

where

$$\kappa(\tau) = \begin{cases} 0.0452 & \text{if } 7.5 \leq \tau < 12.5, \\ 0 & \text{otherwise.} \end{cases} \tag{31}$$

In order to obtain the feasible reference $\mathbf{y}^{\mathrm{r}} = (\mathbf{x}^{\mathrm{r}}, \mathbf{u}^{\mathrm{r}})$, we linearize model (23) to obtain an LTV system and approximate the infinite horizon Problem (8) with a prediction horizon of $M = 600$ and sampling time $t_{\mathrm{s}} = 0.05$ s. For the closed-loop simulations, we use the control input obtained from formulations (7) and (13) with horizon $N = 10$ and sampling time $t_{\mathrm{s}} = 0.05$ s.

Fig. 5 shows the closed-loop trajectories for the initial condition $t_0 = 0$ and $\mathbf{x}_0 = [0.1,\ 0.02,\ 0,\ 50/3.6]^\top$, where the gray lines denote the infeasible reference $\mathbf{r}$, and the black lines denote the optimal reference $\mathbf{y}^{\mathrm{r}}$ from (8). The blue lines show the closed-loop evolution of each state for *ideal* MPC formulation (13), i.e., when the terminal conditions are based on the *feasible* reference $\mathbf{y}^{\mathrm{f}} = \mathbf{y}^{\mathrm{r}}$. The orange lines, on the other hand, show the closed-loop trajectories for the practical MPC formulation (7), which has the terminal conditions based on the infeasible reference, i.e., $\mathbf{y}^{\mathrm{f}} = \mathbf{r}$. The bottom right plot of Fig. 5 shows the closed-loop error with respect to $\mathbf{y}^{\mathbf{r}}$ for the two MPC formulations. It is visible that for times $t \leq 5$ s, the reference trajectory is feasible and both formulations manage to stabilize towards the optimal trajectory. For 5 s $\leq t \leq 15$ s, the discontinuity of the reference $\mathbf{r}$ affects how the two formulations behave. The *ideal* formulation manages to track the optimal reference $\mathbf{y}^{\mathrm{r}}$ (black trajectory) since it is formulated with proper terminal conditions. The practical formulation (orange trajectory) on the other hand tries to track the infeasible reference $\mathbf{r}$, and therefore deviates from the optimal trajectory and ideal

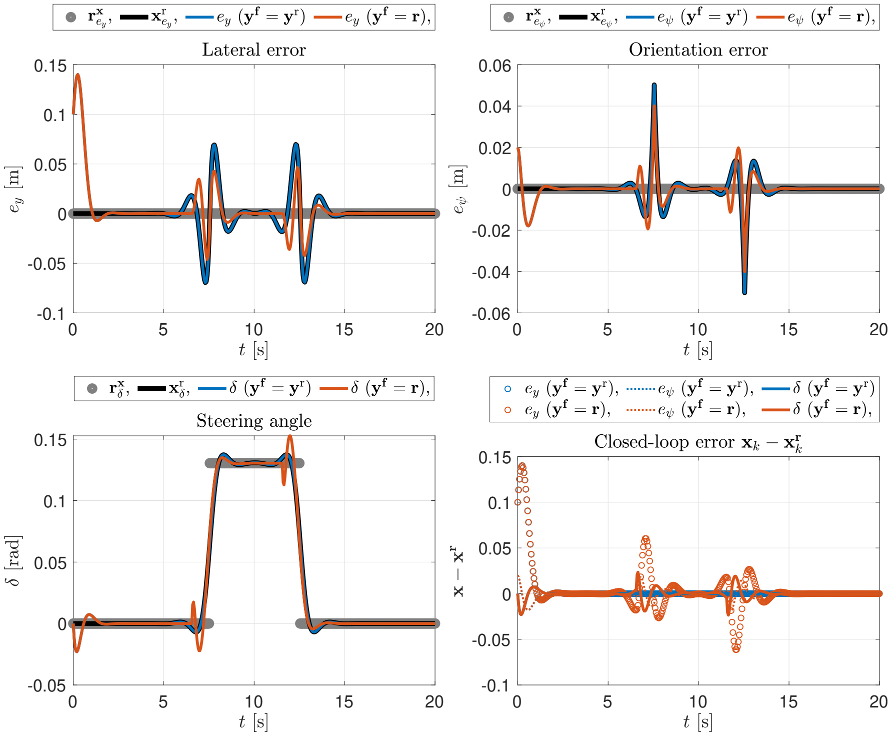

Fig. 5 Closed-loop simulation with initial conditions $\mathbf{x}_0 = [0.1,\ 0.02,\ 0,\ 50/3.6]^\top$ and initial time $t_0 = 0$. The gray trajectories show the infeasible reference $\mathbf{r} = (\mathbf{r}^{\mathbf{x}}, \mathbf{r}^{\mathbf{u}})$, while the black trajectories show the optimal trajectory $\mathbf{y}^{\mathrm{r}} = (\mathbf{x}^{\mathrm{r}}, \mathbf{u}^{\mathrm{r}})$ obtained from Problem (8). The orange trajectories show the closed-loop behavior for the practical MPC Problem (7), while the blue trajectories show the closed-loop behavior for the *ideal* MPC Problem (13).

formulation. After the discontinuity, the rest of the reference trajectory is feasible, and both formulations are asymptotically stabilizing.

In order to verify in simulation the theoretical result of [21, 54] stating that the effect of the terminal condition on the closed-loop trajectory decreases as the prediction horizon N increases, we ran the same simulation with $N = 60$ and observed that the closed-loop trajectory obtained with the infeasible reference becomes indistinguishable from the optimal one by eye inspection.

While we have shown how an infeasible reference affects the closed-loop system in section, we consider next a setting where the reference trajectory becomes infeasible due to pop-up obstacles in the environment. To that end, in the next section we show recursive feasibility guarantees from Theorem 3 are enforced.

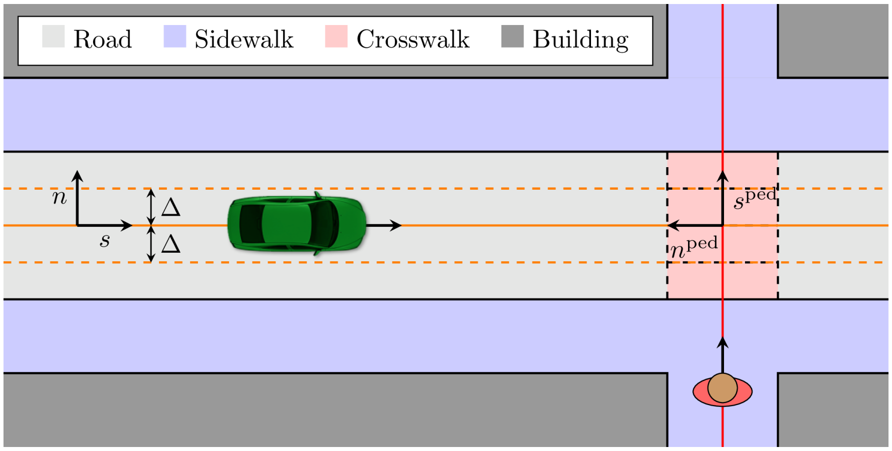

Fig. 6 The simulation setting includes a vehicle driving down a road with a crosswalk where pedestrians might cross.

6.2 MPFTC: Ensuring Safety of the Controller

In this section we illustrate the benefits of the safe terminal set by considering an urban driving environment. In particular, we consider the setting shown in Fig. 6, where the vehicle needs to safely navigate the road and avoid collisions with pedestrians that may be occluded by the environment. To that end, we will show next that when the conditions required our safe framework, i.e., Theorem 3, are satisfied, we are able to avoid collisions with suddenly appearing road users.

For simplicity and ease of illustration, we consider the following reference trajectory

$$\mathbf{r}^{\mathbf{x}}(\tau) = [v^{\text{ref}}\tau,\ 0,\ 0,\ 0,\ v^{\text{ref}}]^\top,\ \mathbf{r}^{\mathbf{u}}(\tau) = [0,\ 0]^\top, \tag{32}$$

which models a constant velocity trajectory with no turning, i.e., the road center line from Fig. 6. In order show constraint satisfaction of our framework while tracking this reference, we need to introduce a model that predicts the future motion of other road users. Hence, to simplify the analysis and presentation of the results, we consider only pedestrians that may appear from the bottom side of the crosswalk in Fig. 6. Therefore, we model the pedestrian dynamics as following the red straight line in Fig. 6. By defining the uncertain variable related to the pedestrian position as $\mathbf{w}_k := [s_k^{\text{ped}}, n_k^{\text{ped}}]$, we formulate the pedestrian kinematics as two single integrators

$$\mathbf{w}_{k+1} = \omega(\mathbf{w}_k, \boldsymbol{\zeta}) = \begin{bmatrix} 1 & 0 \\ 0 & K \end{bmatrix} \mathbf{w}_k + \begin{bmatrix} 1.3 \\ 0 \end{bmatrix} t_{\text{s}} + t_{\text{s}}\boldsymbol{\zeta}, \tag{33}$$

where the lateral states n_k^{ped} are stabilized if $K < 1$, and $\boldsymbol{\zeta} \in \mathcal{Z}$, with $\mathcal{Z} = \{\boldsymbol{\zeta} \mid \|\boldsymbol{\zeta}\|_\infty \leq 0.75\}$, relates to some bounded uncertainty in the pedestrian movement. We stress

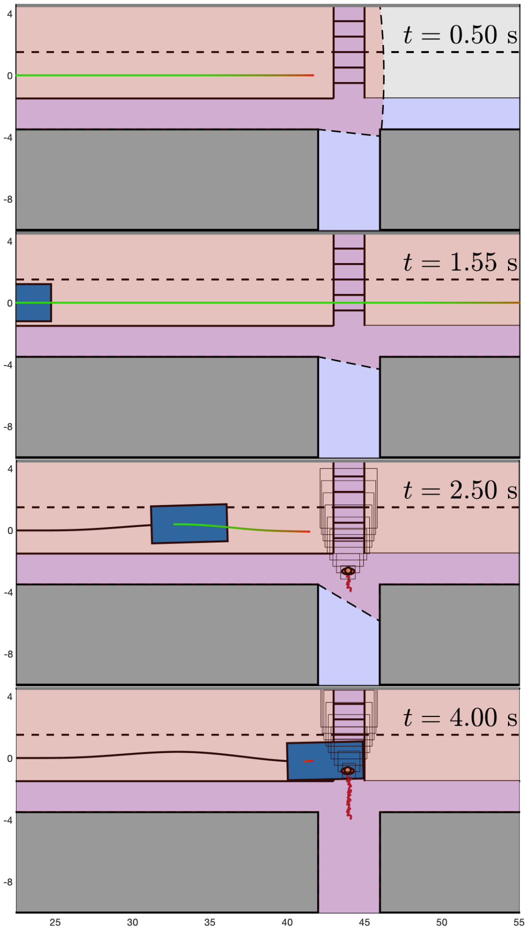

Fig. 7 Four different time instances of the simulation environment. The two top panels show that the sensors (shaded region) cannot see behind a wall, and that the vehicle as such plans a trajectory within the sensing range. The two last panels show that a pedestrian, who was not visible for the sensors, shows up and forces the vehicle to perform an emergency braking.

that we use this model for simplicity, and for real autonomous driving scenarios, model (33) is far from realistic and one should instead consider more advanced pedestrian models like the ones proposed in, e.g., [7, 35, 36, 42].

With model (33), it is straightforward to follow the steps from (18) in Section 5 and propagate the future uncertainty from an initial measurement $\mathbf{w}_k$ as

$$\mathcal{W}_{n+1|k} = \{\omega(\mathbf{w}_{n|k}, \zeta) \mid \mathbf{w}_{n|k} \in \mathcal{W}_{n|k}, \forall \zeta \in \mathcal{Z}\}, \tag{34}$$

with $\mathcal{W}_{k|k} = \mathbf{w}_k$. However, we note that even though prediction model (34) ensures that the predicted sets satisfy $\mathcal{W}_{n|k+1} \subseteq \mathcal{W}_{n|k}$, we are still faced with the problem

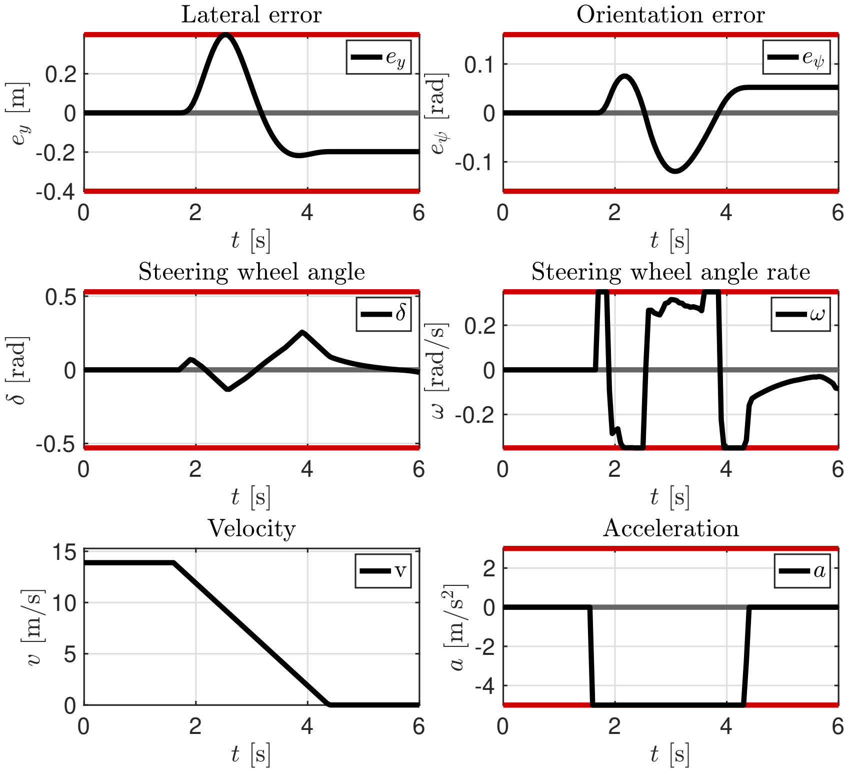

Fig. 8 Closed-loop evolution of the unsafe MPC controller shown in Figure 7. Just before $t \leq 2$ s, a pedestrian appears and forces the vehicle to perform an emergency braking.

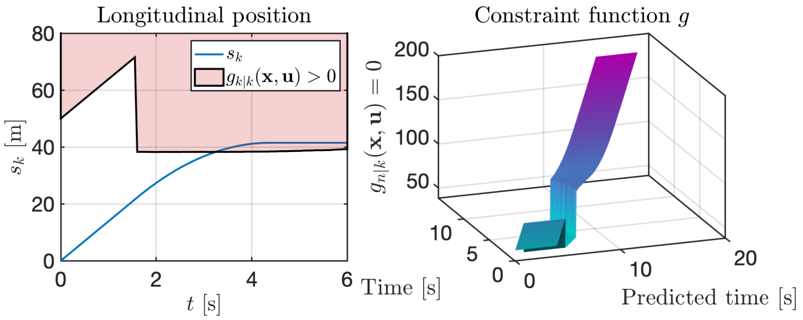

Fig. 9 Constraint evolution for the unsafe MPC controller. The left-hand side shows the constraint $g_{k|k}$, while the right-hand side shows the time evolution of the predicted constraint $g_{n|k}$.

of accounting for pedestrians that are outside of our sensing range, e.g., occluded by buildings or other vehicles. To that end, the sensor-suite must provide information, or have an understanding, where pedestrians might appear. This becomes a crucial part in satisfying Assumption 6, which is needed in Theorem 3 for recursive feasibility. We therefore assume for this simple example, that the sensor-suite has access to information on locations where pedestrians might appear, e.g., from a map. With the information of each such "hidden" location, we expect that a pedestrian might be hidden, and as such, propagate the uncertainty model of (34) for the hidden pedestrian as well. To that end, we must propagate the uncertainty for all measured pedestrians, and potentially "hidden" pedestrians. Hence, if we can measure i pedestrians, and have j hidden locations, we predict the sets $\mathbf{w}^i_{n|k} \in \mathcal{W}^z_{n|k}$, $\forall z \in \mathbb{I}_1^{i+j}$, and construct $g_{n|k}$ as

$$g_{n|k}(\mathbf{x}_{n|k}, \mathbf{u}_{n|k}) = \max_{\mathbf{w}^i_{n|k} \in \mathcal{W}^i_{n|k},\ \forall i \in \mathbb{I}_1^{i+j}} \begin{cases} s_{n|k} + n^{\mathrm{ped},i}_{n|k} + s^{\mathrm{inter}} + r & \text{if } \|s^{\mathrm{ped},i}_{n|k}\| \leq \Delta \\ 0 & \text{otherwise} \end{cases} \leq 0,$$

where s^{inter} is the intersecting point of the reference trajectory and the walkable path, i.e., $(s, n) = (s^{\mathrm{inter}}, 0)$ and $(s^{\mathrm{ped}}, n^{\mathrm{ped}}) = (0, 0)$ map to the same point in the global frame, r is an additional safety distance, and Δ denotes the distance threshold when the pedestrian should be considered for collision avoidance.

Having formulated the collision-avoidance constraints, we now turn to formulating the safe terminal set. In a similar light to Example 1 in Section 5, we consider the safe set to be given when the vehicle is fully stopped, i.e.,

$$\mathcal{X}_{\mathrm{safe}}(\tau) = \{\mathbf{x}\ |v = 0\}. \tag{35}$$

We note that this may not be a suitable safe set for general autonomous driving settings, and that it in general can vary for different cases. However, in order to avoid further technicalities, we consider this safe set to be sufficient for the simplified setting that we use for illustration.

For the simulations we use the terminal set defined in (20), using the sets (35) and (29), with $N = 40$ and $M = 80$. We use the MPFTC formulation (4) with sampling time $t_{\mathrm{s}} = 0.05$ s and nonlinear system model (23). In order to illustrate the benefits of our proposed safe framework, we implement two controllers: one that satisfies Theorem 3; and one where Assumption 6 is not satisfied.

Fig. 7 shows four time instances of the MPC controller which only reacts to what it can sense directly, i.e., the controller does not satisfy Assumption 6. It is visible that the vehicle believes that it can safely cross the intersection in the first two frames. In the last two frames the vehicle has moved close enough to the intersection, such that the sensors can now detect the pedestrian. However, in this case the vehicle velocity is too high, so that it causes a collision with the pedestrian. The closed-loop trajectories are shown in Fig, 8. Here it is visible that the vehicle sees the pedestrian just before $t < 2$ s, and applies full braking. From the lateral error, and orientation error, we realize that the MPC controller in fact tries to avoid a collision, by marginally maximizing the traveled distance by actively steering. Fig. 9 shows that Assumption 6 is not satisfied since constraint $g_{n|k+1} \nleq g_{n|k}$ for all $n > k$, which indeed is needed for safety, i.e., guaranteeing that Theorem 3 holds. The left-hand side shows the constraints at each time k in closed-loop, where it is visible that just

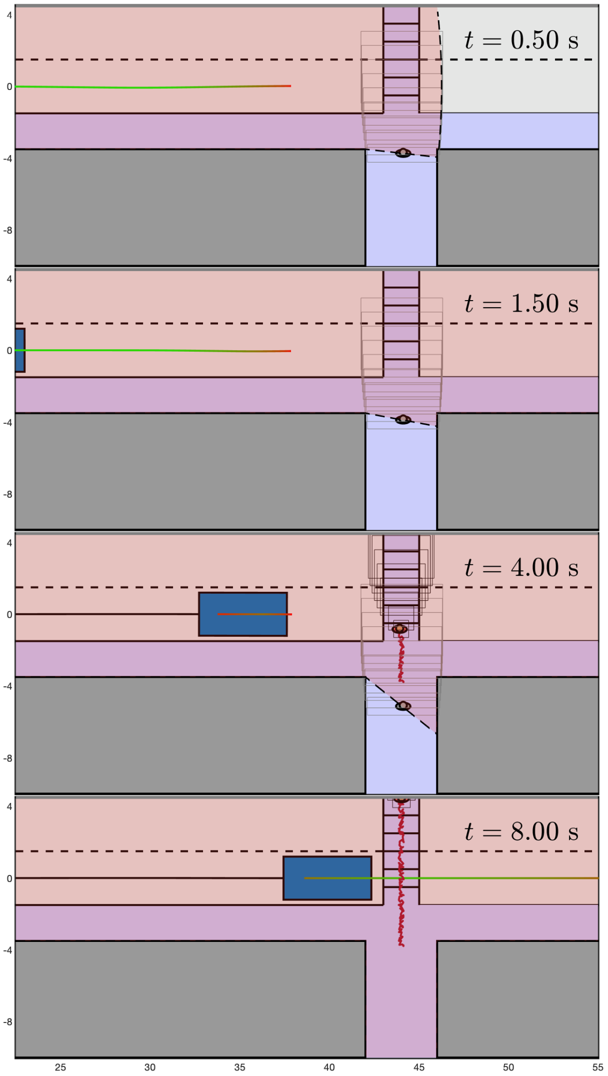

Fig. 10 Four different time instances of the simulation environment. The two top panels show that the sensors (shaded region) cannot see behind a wall, however, the vehicle plans a trajectory as if there were a pedestrian behind the corner. The two last panels show that a pedestrian, who was not visible for the sensors, shows up. Since the vehicle was already prepared for this situation, it manages safely adjust it speed and yield to the pedestrian.

before $t < 2$ s the constraint shrinks, which causes the vehicle to collide with the pedestrian.

Fig. 10 shows how the safe MPC controller behaves when Assumption 6 is satisfied. As opposed to the unsafe controller shown in Fig. 7, the safe controller approaches the intersection more cautiously, as any rational human driver would

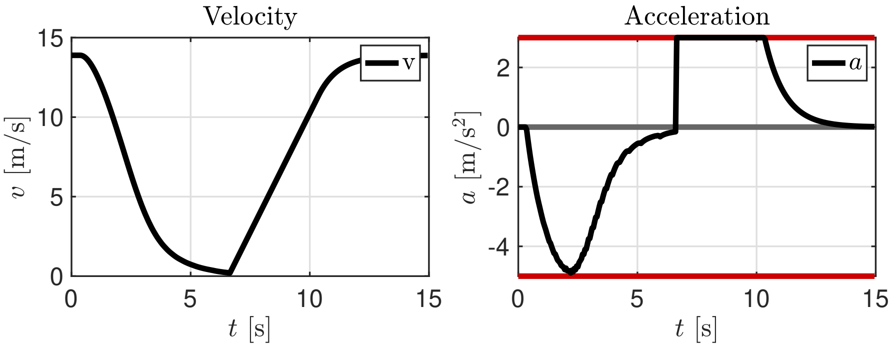

Fig. 11 Closed-loop evolution of the safe MPC controller shown in Fig. 10. The vehicle approaches the intersection by reducing its velocity. After $t = 6$ s the pedestrian passes, and the vehicle is free to accelerate again.

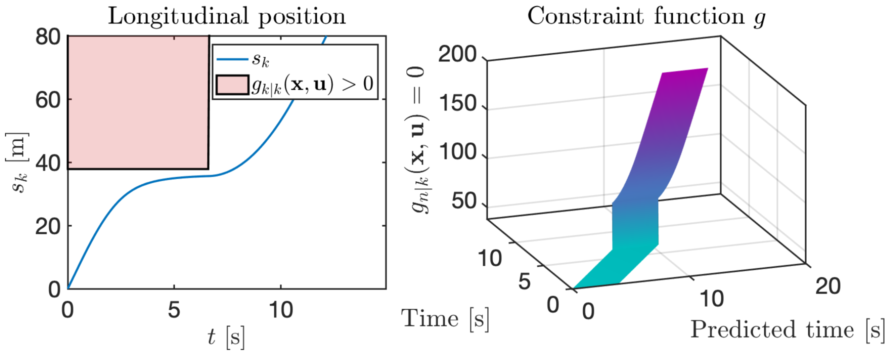

Fig. 12 Constraint evolution for the safe MPC controller in Fig. 10. The left-hand side shows the constraint $g_{k|k}$, while the right-hand side shows the time evolution of the predicted constraint $g_{n|k}$.

do. By adjusting the speed, it anticipates that a moving obstacle may appear behind the wall. This can be seen in the two last frames. As time moves on, the pedestrian can safely pass, and the vehicle moves close enough to see that there are no more remaining pedestrians, so that it can safely accelerate to pass the intersection. Fig. 11 shows the closed-loop velocity and acceleration trajectories. It is worth noting how the vehicle slows down earlier than the unsafe controller in Fig. 8, since it anticipates that a pedestrian might appear behind the corner. Note that, the states (e_y, e_ψ, δ) have been omitted since their dynamics essentially remain unchanged. Finally, Fig. 10 shows that the constraint g is monotonic, and hence, Assumption 6 is satisfied. The "jump" in the right plot illustrates the point in time when the pedestrian is no longer predicted to block the intersection, and the vehicle is free to accelerate again.

7 Conclusions

The possibility of using *infeasible reference trajectories* is of great interest in MPC-based motion planning and control algorithms, due to the convenience and simplicity they offer. In this chapter, we have discussed how such reference trajectories affect the closed-loop behavior of the system, and proposed conditions sufficient for stability to hold. While these results are currently limited to LTV systems, future research will investigate the possibility to also include general nonlinear systems.

Furthermore, we have discussed safety for autonomous driving in a general sense and presented a new safe MPC framework that enables recursive collision-avoidance constraint satisfaction at all times, while relying on assumptions that can be verified in practice on the perception system. Ongoing research is focusing on the practical real-time implementation of the framework in a full-scale test vehicle.

References

1. Amrit, R., Rawlings, J., Angeli, D.: Economic optimization using model predictive control with a terminal cost. Annual Reviews in Control **35**, 178–186 (2011)
2. Andersson, J.A., Gillis, J., Horn, G., Rawlings, J.B., Diehl, M.: Casadi: a software framework for nonlinear optimization and optimal control. Mathematical Programming Computation **11**(1), 1–36 (2019)
3. Batkovic, I., Ali, M., Falcone, P., Zanon, M.: Model predictive control with infeasible reference trajectories. IEEE Transactions on Automatic Control (2020). (submitted, available on arXiv:2109.04846)
4. Batkovic, I., Ali, M., Falcone, P., Zanon, M.: Safe trajectory tracking in uncertain environments. IEEE Transactions on Automatic Control (2020). (submitted, available on arXiv:2001.11602)
5. Batkovic, I., Rosolia, U., Zanon, M., Falcone, P.: A robust scenario mpc approach for uncertain multi-modal obstacles. IEEE Control Systems Letters **5**(3), 947–952 (2020)
6. Batkovic, I., Zanon, M., Ali, M., Falcone, P.: Real-time constrained trajectory planning and vehicle control for proactive autonomous driving with road users. In: Proc. Eur. Control Conf.(ECC) (2019)
7. Batkovic, I., Zanon, M., Lubbe, N., Falcone, P.: A computationally efficient model for pedestrian motion prediction. In: 2018 European Control Conference (ECC), pp. 374–379 (2018). DOI 10.23919/ECC.2018.8550300
8. Blanchini, F., Miani, S.: Set-theoretic methods in control. Springer (2008)
9. Borrelli, F., Bemporad, A., Morari, M.: Predictive control for linear and hybrid systems. Cambridge University Press (2017)
10. Braso, G., Leal-Taixe, L.: Learning a neural solver for multiple object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
11. de Campos, G.R., Falcone, P., Hult, R., Wymeersch, H., Sjöberg, J.: Traffic coordination at road intersections: Autonomous decision-making algorithms using model-based heuristics. IEEE Intelligent Transportation Systems Magazine **9**(1), 8–21 (2017)
12. Campos, G.R., Falcone, P., Wymeersch, H., Hult, R., Sjöberg, J.: Cooperative receding horizon conflict resolution at traffic intersections. In: 53rd IEEE Conference on Decision and Control, pp. 2932–2937. IEEE (2014)
13. Cesari, G., Schildbach, G., Carvalho, A., Borrelli, F.: Scenario model predictive control for lane change assistance and autonomous driving on highways. IEEE Intelligent transportation systems magazine **9**(3), 23–35 (2017)

14. Chen, Y., Rosolia, U., Ubellacker, W., Csomay-Shanklin, N., Ames, A.D.: Interactive multi-modal motion planning with branch model predictive control. arXiv preprint arXiv:2109.05128 (2021)
15. Danelljan, M., Gool, L.V., Timofte, R.: Probabilistic regression for visual tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
16. De Schutter, J., Zanon, M., Diehl, M.: TuneMPC - A Tool for Economic Tuning of Tracking (N)MPC Problems. IEEE Control Systems Letters **4**(4), 910–915 (2020). URL https://github.com/jdeschut/tunempc
17. Diehl, M., Amrit, R., Rawlings, J.B.: A lyapunov function for economic optimizing model predictive control. IEEE Transactions on Automatic Control **56**(3), 703–707 (2010)
18. Faulwasser, T., Findeisen, R.: Nonlinear model predictive control for constrained output path following. IEEE Transactions on Automatic Control **61**(4), 1026–1039 (2016)
19. Faulwasser, T., Grüne, L., Müller, M.: Economic nonlinear model predictive control: Stability, optimality and performance. Foundations and Trends in Systems and Control **5**(1), 1–98 (2018). DOI 10.1561/2600000014
20. Faulwasser, T., Kern, B., Findeisen, R.: Model predictive path-following for constrained nonlinear systems. In: Proceedings of the 48th IEEE Conference on Decision and Control. CDC, 2009., pp. 8642–8647. IEEE (2009)
21. Faulwasser, T., Zanon, M.: Asymptotic Stability of Economic NMPC: The Importance of Adjoints. In: Proceedings of the IFAC Nonlinear Model Predictive Control Conference (2018)
22. Frison, G., Diehl, M.: Hpipm: a high-performance quadratic programming framework for model predictive control. IFAC-PapersOnLine **53**(2), 6563–6569 (2020)
23. González, D., Pérez, J., Milanés, V., Nashashibi, F.: A review of motion planning techniques for automated vehicles. IEEE Transactions on Intelligent Transportation Systems **17**(4), 1135–1145 (2015)
24. Gros, S., Zanon, M., Quirynen, R., Bemporad, A., Diehl, M.: From linear to nonlinear mpc: bridging the gap via the real-time iteration. International Journal of Control **93**(1), 62–80 (2020)
25. Grüne, L.: Economic receding horizon control without terminal constraints. Automatica **49**, 725–734 (2013)
26. Grüne, L., Pannek, J.: Nonlinear Model Predictive Control. Springer, London (2011)
27. Gupta, A., Köroglu, H., Falcone, P.: Computation of low-complexity control-invariant sets for systems with uncertain parameter dependence. Automatica **101**, 330–337 (2019)
28. Gupta, A., Mejari, M., Falcone, P., Piga, D.: Computation of parameter dependent robust invariant sets for lpv models with guaranteed performance. arXiv preprint arXiv:2009.09778 (2020)
29. Gutjahr, B., Gröll, L., Werling, M.: Lateral vehicle trajectory optimization using constrained linear time-varying mpc. IEEE Transactions on Intelligent Transportation Systems **18**(6), 1586–1595 (2017). DOI 10.1109/TITS.2016.2614705
30. Herceg, M., Kvasnica, M., Jones, C., Morari, M.: Multi-Parametric Toolbox 3.0. In: Proc. of the European Control Conference, pp. 502–510. Zürich, Switzerland (2013). http://control.ee.ethz.ch/ mpt
31. Hult, R., Zanon, M., Gros, S., Falcone, P.: Energy-Optimal Coordination of Autonomous Vehicles at Intersections. In: 2018 European Control Conference (ECC), pp. 602–607 (2018)
32. Hult, R., Zanon, M., Gros, S., Falcone, P.: Optimal coordination of automated vehicles at intersections: Theory and experiments. IEEE Transactions on Control Systems Technology **27**(6), 2510–2525 (2018)
33. Johnander, J., Danelljan, M., Brissman, E., Khan, F.S., Felsberg, M.: A generative appearance model for end-to-end video object segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
34. Kerrigan, E.C.: Robust constraint satisfaction: Invariant sets and predictive control. Ph.D. thesis, University of Cambridge (2001)
35. Koschi, M., Althoff, M.: Set-based prediction of traffic participants considering occlusions and traffic rules. IEEE Transactions on Intelligent Vehicles **6**(2), 249–265 (2020)

36. Koschi, M., Pek, C., Beikirch, M., Althoff, M.: Set-based prediction of pedestrians in urban environments considering formalized traffic rules. In: 2018 21st international conference on intelligent transportation systems (ITSC), pp. 2704–2711. IEEE (2018)
37. Lima, P.F., Martensson, J., Wahlberg, B.: Stability conditions for linear time-varying model predictive control in autonomous driving. In: 2017 IEEE 56th Annual Conference on Decision and Control (CDC), pp. 2775–2782. IEEE (2017)
38. Lima, P.F., Pereira, G.C., Martensson, J., Wahlberg, B.: Experimental validation of model predictive control stability for autonomous driving. Control Engineering Practice **81**, 244–255 (2018)
39. Müller, M.A., Angeli, D., Allgöwer, F.: On necessity and robustness of dissipativity in economic model predictive control. IEEE Transactions on Automatic Control **60**(6), 1671–1676 (2015)
40. Nair, S.H., Govindarajan, V., Lin, T., Meissen, C., Tseng, H.E., Borrelli, F.: Stochastic mpc with multi-modal predictions for traffic intersections. arXiv preprint arXiv:2109.09792 (2021)
41. Paden, B., Čáp, M., Yong, S.Z., Yershov, D., Frazzoli, E.: A survey of motion planning and control techniques for self-driving urban vehicles. IEEE Transactions on intelligent vehicles **1**(1), 33–55 (2016)
42. Pek, C., Manzinger, S., Koschi, M., Althoff, M.: Using online verification to prevent autonomous vehicles from causing accidents. Nature Machine Intelligence **2**(9), 518–528 (2020)
43. Rawlings, J., Bonne, D., Jorgensen, J., Venkat, A., Jorgensen, S.: Unreachable Setpoints in Model Predictive Control. IEEE Transactions on Automatic Control **53**, 2209–2215 (2008)
44. Rawlings, J.B., Mayne, D.Q., Diehl, M.: Model predictive control: theory, computation, and design, vol. 2. Nob Hill Publishing Madison, WI (2017)
45. Shalev-Shwartz, S., Shammah, S., Shashua, A.: On a formal model of safe and scalable self-driving cars (2018)
46. Stenborg, E.: Long term localization for self driving cars. Doktorsavhandlingar vid Chalmers tekniska högskola. Ny serie: 4844. Chalmers University of Technology (2020)
47. Stenborg, E., Hammarstrand, L.: Using a single band gnss receiver to improve relative positioning in autonomous cars. In: 2016 IEEE Intelligent Vehicles Symposium (IV), pp. 921–926. IEEE (2016)
48. Uebel, S., Murgovski, N., Bäker, B., Sjöberg, J.: A two-level mpc for energy management including velocity control of hybrid electric vehicles. IEEE Transactions on Vehicular Technology **68**(6), 5494–5505 (2019)
49. Verschueren, R., Frison, G., Kouzoupis, D., Frey, J., van Duijkeren, N., Zanelli, A., Novoselnik, B., Albin, T., Quirynen, R., Diehl, M.: acados: a modular open-source framework for fast embedded optimal control (2020)
50. Wächter, A., Biegler, L.T.: On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Mathematical programming **106**(1), 25–57 (2006)
51. Yu, S., Maier, C., Chen, H., Allgöwer, F.: Tube mpc scheme based on robust control invariant set with application to lipschitz nonlinear systems. Systems & Control Letters **62**(2), 194–200 (2013)
52. Zanon, M.: A gauss-newton-like hessian approximation for economic nmpc. IEEE Transactions on Automatic Control (2020)
53. Zanon, M.: A Gauss-Newton-Like Hessian Approximation for Economic NMPC. IEEE Transactions on Automatic Control **66**(9), 4206–4213 (2021)
54. Zanon, M., Faulwasser, T.: Economic MPC without terminal constraints: Gradient-correcting end penalties enforce asymptotic stability. Journal of Process Control **63**, 1 – 14 (2018)
55. Zanon, M., Gros, S., Diehl, M.: A Lyapunov Function for Periodic Economic Optimizing Model Predictive Control. In: Proceedings of the 52nd Conference on Decision and Control (CDC), pp. 5107–5112 (2013)
56. Zanon, M., Gros, S., Diehl, M.: Indefinite Linear MPC and Approximated Economic MPC for Nonlinear Systems. Journal of Process Control **24**, 1273–1281 (2014)
57. Zanon, M., Gros, S., Diehl, M.: A Tracking MPC Formulation that is Locally Equivalent to Economic MPC. Journal of Process Control **45**, 30 – 42 (2016)

58. Zanon, M., Gros, S., Diehl, M.: A Periodic Tracking MPC that is Locally Equivalent to Periodic Economic MPC. In: Proceedings of the 2017 IFAC World Congress, vol. 50(1), pp. 10711–10716 (2017)
59. Zanon, M., Grüne, L., Diehl, M.: Periodic optimal control, dissipativity and MPC. IEEE Transactions on Automatic Control **62**(6), 2943–2949 (2017)