2018 21st International Conference on Intelligent Transportation Systems (ITSC)
Maui, Hawaii, USA, November 4-7, 2018

Supervised Trajectory Tracking Control

Ricardo de Castro and Jonathan Brembeck

Abstract— **This work deals with the design of a supervisory module for the motion control of autonomous vehicles. The supervisor's goal is to monitor the behavior of a trajectory tracking controller (TTC), introducing corrective control actions to prevent violations of safety and input constraints. It relies on the combination of control barrier functions and optimization methods, and considers Lyapunov-based constraints and costs to penalize performance degradation due to supervisor's corrective actions. Simulation results of a collision avoidance scenario demonstrate the effectiveness of the proposed supervised TTC.**

I. Introduction

Autonomous driving is seen as a promising technology that could increase overall transportation efficiency, reduce road accidents and improve comfort of future mobility systems. For the successful deployment of this technology several technical areas need to be addressed, including reliable perception of the environment, safe trajectory generation and accurate vehicle control. This paper focuses on the vehicle motion controller. Because of non-linear vehicle dynamics, uncertain parameters (e.g. the tire-road friction coefficient), and input and state constraints, designing the motion controller is a challenging task [1].

To cope with these challenges, several types of motion controllers have been investigated in the literature. They can be classified based on control objective (e.g., path stabilization *vs* trajectory stabilization), vehicle modeling assumptions and control method. The objective of trajectory stabilization is to follow a time parameterized reference, e.g. geometric path with an associated timing law, while in the path stabilization a time-independent geometric path is tracked with a pre-defined velocity [1]. Since the former has stricter time constraints, it is normally preferable for time-critical tasks, such as emergency maneuvers [2], and is adopted in this work. In terms of motion control methods, kinematic-based controllers [3] are simple to implement and numerically well behaved at standstill. However, their performance degrades when the vehicle significantly deviates from pure-rolling assumptions [2]. To overcome this issue, motion controllers based on dynamic vehicle models, which take into account inertial and tire-slippage effects, are adopted. They can be designed based on a wide variety of methods, including linear control [4], immersion and invariance (I&I) [5], flatness [6], input-output linearization [7], sliding modes [2], and optimization-based controllers such as model-predictive control (MPC) [8]. As discussed in recent benchmarking studies [2], [5], these methods offer robustness against model uncertainty (sliding modes), good noise rejection (I&I), prediction and constraint handling capabilities (MPC), but can also suffer from sensitivity to parameter uncertainties (I&I), high computation effort (MPC) or chattering (sliding modes).

With the exception of MPC-based methods, the majority of these previous contributions focus on *unconstrained motion control* problems, where input and state constraints are neglected in the theoretical design of the control law. Such constraints can appear due to actuation saturations and force limits in the tire-road interface, e.g, friction ellipse constraints [9], and are critical when controlling the vehicle close to is physical limits. If not properly handled, these constraints can negatively affect the motion control performance, increase tracking errors and potentially compromise the safe execution of planned maneuvers.

Motivated by this issue, a novel supervisory module for the trajectory tracking task is proposed in this work. The main contribution of the supervisor is to endow a nominal trajectory tracking controller (TTC) - designed around an *unconstrained motion control* law - with constraint handling capabilities. Specifically, the supervisor produces corrective control actions to enforce actuation and physical constraints, while minimizing the performance degradation of the nominal TTC. The supervisor also enhances the nominal TTC with safety features, such as collision avoidance and fulfillment of vehicle stability limits. These safety features, which represent the second main contribution of the supervisor, are relevant for several reasons. For instance, when the planned trajectory is computed remotely, e.g. using a virtual trajectory planner running on the cloud, the (local) supervisor can correct the reference trajectory to cope with unexpected events not considered in the remote computations. The supervisor can also be seen as a fail-safe mechanism, allowing the vehicle to safely operate in case of faults in the planner module.

II. Model and Problem Formulation

This section presents the control-oriented model of the (ego) vehicle and mathematically defines the trajectory tracking control problem. The kinematic component of the vehicle model is defined as:

$$\begin{bmatrix}\dot{p}\\ \dot{\psi}\end{bmatrix} = R(\psi)v, \quad R(\psi) = \begin{bmatrix}\cos(\psi) & -\sin(\psi) & 0\\ \sin(\psi) & \cos(\psi) & 0\\ 0 & 0 & 1\end{bmatrix} \tag{1}$$

where $p = \begin{bmatrix}p_X & p_Y\end{bmatrix}^T$ represents the position of the vehicle's center of gravity – defined in a reference axis fixed with road surface – and ψ the yaw-angle (see Fig. 1). The velocity

The authors are with Institute of System Dynamics and Control, German Aerospace Center (DLR), Münchner Strasse 20, 82234 Wessling, Germany Ricardo.DeCastro@dlr.de

978-1-7281-0322-8/18/$31.00 ©2018 IEEE

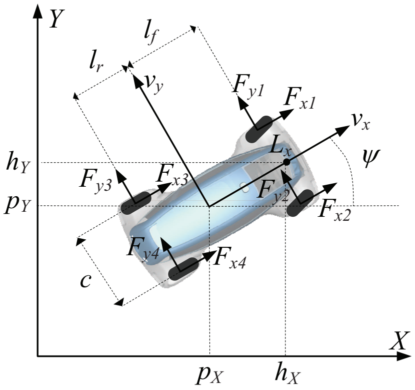


Fig. 1. Planar vehicle model. The point (p_X, p_Y) represents the position and orientation of the vehicle's center of gravity.

vector $v = \begin{bmatrix} v_x & v_y & v_\psi \end{bmatrix}^T$ is composed of longitudinal (v_x), lateral (v_y), and yaw (v_ψ) velocity, respectively, defined in a reference frame fixed with the vehicle's center of gravity. Its dynamics are given by [4]

$$\begin{aligned} M\dot{v} &= Mf_v(v) + BF_{xy}, \quad f_v(v) = \begin{bmatrix} v_y v_\psi & -v_x v_\psi & 0 \end{bmatrix}^T && (2) \\ B &= \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ -\frac{c}{2} & l_f & \frac{c}{2} & l_f & -\frac{c}{2} & -l_r & \frac{c}{2} & -l_r \end{bmatrix} && (3) \\ F_{xy} &= \begin{bmatrix} F_{x1} & F_{y1} & F_{x2} & F_{y2} & F_{x3} & F_{y3} & F_{x4} & F_{y4} \end{bmatrix} && (4) \end{aligned}$$

where F_{xi} and F_{yi} represent the longitudinal and lateral forces generated by the tire $i \in \{1,2,3,4\}$[1], respectively. The matrices $M = \mathrm{diag}(m, m, I_z)$ and B are dependent on the vehicle mass (m), yaw inertial (I_z), position of center of gravity (l_f, l_r) and trackwidth (c). The vehicle's front wheels are assumed to have independent steering, braking and traction actuators, while the rear wheels are equipped with individual braking actuators. Due to this highly-actuated structure, the front tires forces ($F_{xi}, F_{yi}, i = 1,2$) and the rear longitudinal forces ($F_{xi}, i = 3,4$) are assumed to be directly controlled, $u = \begin{bmatrix} F_{x1} & F_{y1} & F_{x2} & F_{y2} & F_{x3} & F_{x4} \end{bmatrix}$, where $u \in \mathcal{U} \subset \mathbb{R}^{n_u}$ is the control input vector and $n_u = 6$ the number of control variables. These controllable forces are subject to friction ellipse constraints [9] and physical actuation limits, which are captured by the input domain $\mathcal{U}$:

$$\begin{aligned} \mathcal{U} \;=\; \{u \in \mathbb{R}^{n_u} : \quad & \left(\frac{F_{xi}}{d_{xi}}\right)^2 + \left(\frac{F_{yi}}{d_{yi}}\right)^2 \le (\mu_i F_{zi})^2, \; i = 1,2 \\ & \underline{F_{xi}} \le F_{xi} \le \overline{F_{xi}}, \qquad i = 1,2 \\ & \max(-\mu_i F_{zi}, \underline{F_{xi}}) \le F_{xi} \le 0, \qquad i = 3,4 \quad \} && (5) \end{aligned}$$

where μ_i is the tire-road friction coefficient, d_{xi} and d_{yi} constants, and $[\underline{F_{xi}}, \overline{F_{xi}}]$ force limits due to actuation constraints. The vertical loads of the tires, F_{zi}, are considered here as known exogenous signals, which can be estimated using quasi-static mappings between vehicle's accelerations and vertical load transfer [4].

To model the coupling between the lateral and longitudinal forces of the rear tires, the magic tire formula [9] and the friction ellipse constraints are employed:

$$\begin{aligned} F_{yi}(v, F_{xi}) &= \tilde{F}_{yi}(v)\varphi(F_{xi}), \; \varphi(F_{xi}) = \sqrt{1 - \left(\frac{F_{xi}}{\mu_i F_{zi}}\right)^2} \\ \tilde{F}_{yi}(v) &= \mu_i F_{zi} \sin(C_i \,\mathrm{atan}(B_i \alpha_i(v))) && (6) \\ \tan(\alpha_i(v)) &= -\frac{\vartheta_{yi}^T v}{\vartheta_{xi}^T v}, \quad \begin{bmatrix} \vartheta_{xi}^T \\ \vartheta_{yi}^T \end{bmatrix} = \begin{bmatrix} 1 & 0 & \chi_{xi} \\ 0 & 1 & \chi_{yi} \end{bmatrix} && (7) \\ \begin{bmatrix} \chi_{x3} & \chi_{x4} \end{bmatrix} &= \begin{bmatrix} -\frac{c}{2} & \frac{c}{2} \end{bmatrix}, \quad \begin{bmatrix} \chi_{y3} & \chi_{y4} \end{bmatrix} = \begin{bmatrix} -l_r & -l_r \end{bmatrix} \end{aligned}$$

for $i = 3,4$, where B_i, D_i are parameters and $\alpha_i(v)$ the side-slip angle of the tire i. The term $\tilde{F}_{yi}$ represents the lateral force generated by the rear tire i under pure lateral side-slip conditions. It is "derated" by the factor $\varphi(F_{xi}) \in [0,1]$ when longitudinal forces are applied [10]. To facilitate the handling of the force coupling between longitudinal and lateral rear forces, the derating factor $\varphi(F_{xi})$ is approximated by $\varphi(F_{xi}) \approx 1 + \frac{F_{xi}}{\mu_i F_{zi}}$. This pragmatic approximation, valid in the domain $F_{xi} \in [\underline{F_{xi}}, 0]$, $i = 3,4$, allows us to obtain an affine mapping between F_{xy} and control input u:

$$F_{xy} \;\approx\; B_u(v)u + f_y(v) \qquad (8)$$

where $B_u(v)$ and $f_y(v)$ are defined in Appendix. Based on these considerations, the vehicle dynamic model can be compactly described as:

$$\dot{x} = \begin{bmatrix} \dot{p} \\ \dot{\psi} \\ \dot{v} \end{bmatrix} = \underbrace{\begin{bmatrix} R(\psi)v \\ f_v(v) + M^{-1}Bf_y(v) \end{bmatrix}}_{f_x(x)} + \underbrace{\begin{bmatrix} 0 \\ G(x) \end{bmatrix}}_{G_x(x)} u + \underbrace{\begin{bmatrix} 0 \\ I_{3\times3} \end{bmatrix}}_{G_d} d \qquad (9)$$

where $x = \begin{bmatrix} p^T & \psi & v^T \end{bmatrix}^T \in \mathcal{X} \subset \mathbb{R}^6$, $G(x) = M^{-1}BB_u(v)$, $I_{3\times3}$ is an identity matrix (with dimension 3), $\mathcal{X}$ the state domain, and $d \in \mathcal{D} \subset \mathbb{R}^3$ a disturbance acceleration that captures modeling approximation errors - e.g. introduced by (8) - and external perturbations. The disturbance is assumed to be bounded by the set $\mathcal{D} = \{d \in \mathbb{R}^3 : \|d\| \le \overline{d}\}$, where $\overline{d}$ is a known upper bound.

In order to formulate the motion control problem, let us consider a set of safety constrains $\mathcal{C} \subset \mathcal{X}$ and the virtual point h, located at a constant distance L_x from the vehicle's center of gravity (see Fig. 1):

$$h = \begin{bmatrix} h_X & h_Y & h_\psi \end{bmatrix}^T = \begin{bmatrix} p^T & \psi \end{bmatrix}^T + R(\psi)L \qquad (10)$$

where $L = \begin{bmatrix} L_x & 0 & 0 \end{bmatrix}^T$. This point can be selected, e.g., as the center of oscillation ($L_x = I_z/(ml_r)$), which cancels the effect of rear lateral forces F_{yi} in the tracking of h_X, h_Y [6]. Our goal is to construct a feedback control law for $u \in \mathcal{U}$ such that the virtual point h tracks, as close as possible, the reference trajectory $h_r(t) : \mathbb{R} \to \mathbb{R}^3$, while enforcing safety constraints $x \in \mathcal{C}$, $\forall d \in \mathcal{D}$.

The TTC's control law is composed by the summation of two components, $u = u_N + \Delta u$. The first component (u_N) is a nominal control action. It uses input-output (IO) linearization [11] method to compute the control actions for the unconstrained control problem. The second component

[1] as depicted in Fig. 1, index 1 refers to the front left tire, 2 to the front right, 3 to the rear left, and 4 to the rear right tire

(Δu) is a corrective control action generated by the supervisor to cope with the problem's constraints ($\mathcal{U},\mathcal{C}$). In the final step, the commanded tire forces u are translated into actuator demands (steering angles and wheel torques) using model inversion methods (see [6], [4] for details).

III. Nominal Trajectory Tracking Controller

To derive the nominal controller u_N, let us consider the tracking error $e = \begin{bmatrix} e_X & e_Y & e_\psi \end{bmatrix}^T = h - h_r$. Since all elements of e have a relative degree of 2, the tracking error and its derivative can be augmented into the vector $z = \begin{bmatrix} e_X & \dot{e}_X & e_Y & \dot{e}_Y & e_\psi & \dot{e}_\psi \end{bmatrix}^T$, enabling us to express the error dynamics as:

$$\begin{aligned} \dot{z} &= Az + B\Gamma(x, \ddot{h}_r) + BE(x)\,(\tau + d) && (11)\\ \tau &= G(x)u && (12) \end{aligned}$$

where $A, B, \Gamma(x, \ddot{h}_r), E(x)$ are matrices defined in Appendix. The above representation contains the virtual control input $\tau \in \mathbb{R}^{n_\tau}$, $n_\tau = 3$, composed by the longitudinal, lateral and yaw acceleration produced by the actuators. Because of this virtual control input, the nominal control problem can be divided in two steps. In the first step, a high-level controller, based on IO linearization [11], manipulates the virtual input τ in order to fulfill the primary control goal. Its control law is given as

$$\tau_N = E^{-1}(x)\left(-\Gamma(x, \ddot{h}_r) - Kz\right) \tag{13}$$

where $K \in \mathbb{R}^{3\times 6}$ is a matrix selected by the designer such that $(A - BK)$ is Hurwitz. In the second step, the nominal virtual input τ_N is mapped into the control input space $\mathbb{R}^{n_u}$. Because of its numerically efficiency and simplicity, the Moore-Penrose pseudo-inverse $u_N = G(x)^+\tau_N$, with $G(x)^+ = G(x)^T(G(x)G(x)^T)^{-1}$, is employed.

IV. Supervisor

This section presents the main novelty of this work: the supervisor module for the TTC. The supervisor's main goal is to enhance the nominal TTC with constraint handling capabilities. It achieves that by generating a corrective control action $\Delta u \in \mathbb{R}^{n_u}$, fulfilling: *i) input constraints*, $u_N + \Delta u \in \mathcal{U}$, and *ii) safety constraints*, $u_N + \Delta u \in \mathcal{U}_C$, where $\mathcal{U}_C$ is a control input set constructed based on control barrier functions [12] and able to enforce the safety set $\mathcal{C} \subset \mathcal{X}$ (details follow shortly). To comply with these constraints, the supervisor might need to modify the nominal virtual input, i.e.,

$$\tau = \tau_N + \Delta\tau, \quad \Delta\tau = G(x)\Delta u \tag{14}$$

where $\Delta\tau$ is the corrective virtual input due to Δu. Deviations from the nominal virtual input occur when $\Delta\tau \neq 0$ and might negatively affect the performance of the nominal TTC. To mitigate these risks, the supervisor should be *minimally invasive*, i.e., the performance degradation of the nominal TTC should be as small as possible.

A. Performance Degradation

To better understand the *minimally invasive* property, let us treat the supervisor's corrective virtual input $\Delta\tau$ as an additional perturbation and investigate its effect in the closed-loop system using Lyapunov methods. Accordingly, inserting (14) into (11) leads to the following error dynamics:

$$\dot{z} = (A - BK)z + BE(x)(\Delta\tau + d) \tag{15}$$

Given the dominance of the linear component $(A - BK)z$, stability can be investigated using the quadratic Lyapunov function $V(z) = z^TPz$, where P is a positive definite matrix[2]. Computing the time derivative of V one obtains

$$\dot{V}(z, \Delta_{\dot{V}}, d) = -z^TQz + 2z^TPBE(x)d + \Delta_{\dot{V}} \tag{16}$$

where $\Delta_{\dot{V}} = 2z^TPBE(x)\Delta\tau$ represents a perturbation introduced by the supervisor's intervention. Recall that, ultimate boundedness of z – i.e., convergence of z to a small neighborhood around the origin – is achieved if $\dot{V}$ fulfills [11]

$$\dot{V}(z, \Delta_{\dot{V}}, d) \leq -W_V(z) < 0, \qquad \forall \|z\| \geq \varepsilon_z,\ d \in \mathcal{D} \tag{17}$$

where $W_V(z)$ is a positive definite function and $\varepsilon_z > 0$ a small positive constant that affects the size of the ultimate bound. It can be verified that the nominal TTC ensures ultimate boundedness of z, i.e., $\dot{V}(z, 0, d)$ fulfills (17). On the other hand, the supervisor's perturbation, $\Delta_{\dot{V}}$ can have a "good" or a "bad" effect in the ultimate boundedness condition. For example, a "good" effect is obtained when $\Delta_{\dot{V}} < 0$, which makes $\dot{V}$ more negative, helping the fulfillment of (17). "Bad" effects occur when $\Delta_{\dot{V}} > 0$. In this case, ε_z might need to be increased in order to fulfill (17), leading to higher ultimate bounds and – in the worst case scenario – compromise the closed-loop stability. In light of this discussion, the supervisor should penalize "bad" perturbations, i.e., positive values of $\Delta_{\dot{V}}$. This can be achieved, for example, by minimizing $J_s(s) = w_s s^2$ subject to $\Delta_{\dot{V}}(z, x, \Delta\tau) \leq s$ and $s \geq 0$, where s is a non-negative upper bound for $\Delta_{\dot{V}}$ and w_s a positive weight.

B. Overview of the Optimal Supervisor

Constraint fulfillment and minimization of performance degradation can be achieved using the following *optimization-based supervisor*:

$$\begin{aligned} \min_{\Delta u, \Delta\tau, s} \quad & J_s(s) + J_\tau(\Delta\tau) + J_u(u_N + \Delta u) \\ s.t. \quad & u_N + \Delta u \in \mathcal{U} \cap \mathcal{U}_C && (18a)\\ & \Delta\tau = G(x)\Delta u && (18b)\\ & 2z^TPBE(x)\Delta\tau \leq s, \quad s \geq 0 && (18c) \end{aligned}$$

where J_u is a cost term that minimizes a secondary actuation cost, based on the friction usage of the tires (see Appendix), $J_\tau(\Delta\tau) = \Delta\tau^T W_\tau \Delta\tau$, and W_τ a weight matrix. This optimal formulation computes the correction term Δu compliant with $\mathcal{U}$ and $\mathcal{U}_C$, while simultaneously penalizing the performance degradation (J_s and (18c)), deviations of the nominal virtual input (J_τ) and actuation cost (J_u).

[2] obtained by solving the equation $P(A - BK) + (A - BK)^TP = -Q$, with Q a positive definite matrix defined by the designer

C. Safety Constraints

Let us now discuss in more detail how the safety constraints $\mathcal{C}$ and corresponding safe input set $\mathcal{U}_{\mathcal{C}}$ can be designed. To assist us in this discussion, we define the generic safe set $\mathcal{C}(\theta) \subset \mathcal{X}$ as the superlevel set of the scalar function $l(x,\theta) \in \mathbb{R}$:

$$\mathcal{C}(\theta) = \{x \in \mathcal{X} : \quad l(x,\theta) \geq 0 \quad \} \tag{19}$$

where $\theta \in \Theta \subset \mathbb{R}^{n_\theta}$ is a vector of n_θ time-varying parameters. From a technical point of view, forward invariance of the safe set $\mathcal{C}(\theta)$ is desired, i.e., if $x(0) \in \mathcal{C}(\theta)$ then $x(t) \in \mathcal{C}(\theta), \forall t \geq 0$. To ensure this property, control barrier functions - in the form of zeroing control barrier function (ZCBF) - are employed.

Definition 1. *[12] Consider the set $\mathcal{C}(\theta) \subset \mathcal{X} \subset \mathbb{R}^{n_x}$ defined by* (19) *for the function $l(x,\theta) \in \mathbb{R}$. The function $l(x,\theta)$ is a zeroing control barrier function (ZCBF) if there is an extended class K function*[3] *α_b such that $\dot{l}(x,u,d,\theta,\dot{\theta}) \geq -\alpha_b(l)$, $\forall x \in \mathcal{X}, \theta \in \Theta, \dot{\theta} \in \dot{\Theta}, d \in \mathcal{D}$. This is equivalent to*

$$\sup_{u \in \mathcal{U}} \inf_{d \in \mathcal{D}} \Big[\frac{\partial l}{\partial x}(f_x(x) + G_x(x)u + G_d d) + \frac{\partial l}{\partial \theta}\dot{\theta} + \alpha_b(l) \Big] \geq 0,$$

where l is short notation for $l(x,\theta)$.

The key idea of the ZCBF definition is to ensure the existence of a control input $u \in \mathcal{U}$ capable of enforcing the condition $\dot{l} \geq -\alpha_b(l)$. To facilitate the computation of $\mathcal{U}_{\mathcal{C}}$, i.e., the set of control inputs capable of enforcing $\dot{l} \geq -\alpha_b(l)$, we will assume that a lower bound $\underline{\varepsilon}_d(x)$, possibly state dependent, for the disturbance d can be computed, i.e., $\inf_{\theta \in \Theta, d \in \mathcal{D}} \frac{\partial l(x,\theta)}{\partial x} G_d d \geq \underline{\varepsilon}_d(x)$. Based on this bound, a conservative estimate for $\mathcal{U}_{\mathcal{C}}$ can be determined as:

$$\begin{aligned}\mathcal{U}_{\mathcal{C}}(x,\theta,\dot{\theta}) \quad = \quad \{u \in \mathcal{U} : \ & \frac{\partial l(x,\theta)}{\partial x}(f_x(x) + G_x(x)u) + \\ & \frac{\partial l(x,\theta)}{\partial \theta}\dot{\theta} + \underline{\varepsilon}_d(x) + \alpha_b(l(x,\theta)) \geq 0\}\end{aligned}$$

Lemma 1. *[12] Consider the safe set $\mathcal{C}(\theta)$ defined by* (19) *and the ZCBF $l(x,\theta)$. Any Lipschitz continuous controller u such that $u \in \mathcal{U}_{\mathcal{C}}(x,\theta,\dot{\theta}) \neq \emptyset$ for the system* (9) *renders the set $\mathcal{C}(\theta)$ forward invariant.*

We will now apply this result to design ZCBFs for safety constraints related with vehicle stability limits ($\mathcal{C}_{\mathcal{T}}$) and obstacle avoidance ($\mathcal{C}_O$).

1) Vehicle Stability Limits: The nominal TTC does not directly control the rear lateral forces ($F_{yi}, i = 3,4$). When faced with model mismatches and disturbances, these forces can saturate, leading to high side-slip angles, loss of grip and possible vehicle instability [13]. To prevent this loss of stability, the supervisor will constrain the rear side-slip angles to lie in the *safe slip-angle envelope* $\mathcal{C}_{\mathcal{T}}$:

$$\mathcal{C}_{\mathcal{T}} = \{x \in \mathcal{X} : |\tan(\alpha_i(v))| \leq \tan(\overline{\alpha}),\ v = C_v x,\ i = 3,4\}$$

where $\overline{\alpha}$ is maximum allowed side-slip angle. By exploiting the fact that $\tan(\alpha_i(v))$ is a rational function of the vehicle velocity v, enables us to represent this safety set as an intersection of n_T polytopic sets $\mathcal{C}_{\mathcal{T}} = \cap_{j=1}^{n_T} \mathcal{C}_{\mathcal{T},j}$, as described in Algorithm 1. To better understand the construction of these polytopic sets, let us consider one of the upper side-slip bounds in $\mathcal{C}_T$, namely $\tan(\alpha_3(v)) \leq \tan(\overline{\alpha})$. Expanding the tangent function with its rational representation – see (7) – yields

$$\tan(\alpha_3(v)) = -\frac{\vartheta_{y3}^T v}{\vartheta_{x3}^T v} = -\frac{\vartheta_{y3}^T C_v x}{\vartheta_{x3}^T C_v x} \leq \tan(\overline{\alpha}), \tag{20}$$

where C_v is a matrix that extracts v from x. This is equivalent to $\left(\vartheta_{y3}^T + \vartheta_{x3}^T \tan(\overline{\alpha})\right) C_v x \geq 0$, which leads to the polytopic set $\mathcal{C}_{\mathcal{T},1}$ defined in Algorithm 1. The construction of the remaining polytopic sets follows similar arguments.

Algorithm 1 Barrier function for rear tire's protection

```
$\begin{bmatrix} q_1^T \\ q_2^T \\ q_3^T \\ q_4^T \end{bmatrix} = \begin{bmatrix} \vartheta_{y3}^T + \vartheta_{x3}^T \tan(\overline{\alpha}) \\ -\vartheta_{y3}^T - \vartheta_{x3}^T \tan(-\overline{\alpha}) \\ \vartheta_{y4}^T + \vartheta_{x4}^T \tan(\overline{\alpha}) \\ -\vartheta_{y4}^T - \vartheta_{x4}^T \tan(-\overline{\alpha}) \end{bmatrix}$
$n_T = 4$
for $j = 1$ to $n_T$ do
    $\mathcal{C}_{\mathcal{T},j} = \{x \in \mathcal{X} : \ l_{\mathcal{T},j}(x) = q_j^T C_v x \geq 0\}$
    $\mathcal{U}_{\mathcal{C}_{\mathcal{T},j}}(x) = \{u \in \mathcal{U} : -q_j^T C_v G_x(x) u \leq c_j(x)\}$
    $c_j(x) = q_j^T C_v f_x(x) + \underline{\varepsilon}_{d,j} + \alpha_{b,j}(l_{\mathcal{T},j})$
end for{note: $C_v$ extracts $v$ from $x$ , $\underline{\varepsilon}_{d,j} = -\left\|q_j^T C_v G_d\right\| \overline{d}$}
```

From Lemma 1, forward invariance of the *safe slip-angle envelope* $\mathcal{C}_{\mathcal{T}}$ can be achieved as long as $u_N + \Delta u \in \mathcal{U}_{\mathcal{C}_{\mathcal{T}}} = \cap_{j=1}^{n_T} \mathcal{U}_{\mathcal{C}_{\mathcal{T},j}}$.

2) Obstacle Avoidance: The second set of barrier functions deals with collision avoidance between the ego-vehicle and a set of n_o agents (obstacles). Our idea is to extend collision-free ZCBFs, previously proposed by [14] in the context of multirobot systems, to automotive applications. As depicted in Fig. 2, each agent has position p_k, velocity $\dot{p}_k$ and acceleration $\ddot{p}_k$, $k = 1,\ldots,n_o$; these agent states can be obtained using data from vehicle mounted sensors and its accuracy enhanced with inter-vehicle communications. We define safety disks, with radius D_k, around agent k, while the ego-vehicle is protected by a safety disk with radius D_0. A collision with agent k occurs if the relative distance $\Delta p_k = p - p_k$ between the ego-vehicle and agent k is smaller than $D_{s,k} = D_k + D_0$, i.e., $\|\Delta p_k\| \leq D_{s,k}$. To avoid such outcomes, Δp_k should be sufficiently high, such that the relative velocity $\Delta v_k = \dot{p} - \dot{p}_k$ can be driven to zero before a collision happens. To better explain this concept, let us project the relative velocity Δv_k along the tangent unit vector $t_k = \Delta p_k / \|\Delta p_k\|$, leading to the tangential relative velocity $\Delta v_{k,t} = t_k^T \Delta v_k$ (see Fig. 2). As discussed in [14], if

[3] A continuous function $\alpha_b : (-b,a) \longrightarrow \mathbb{R}$ for some $a,b > 0$ is an *extended class K* function if *i)* $\alpha_b(0) = 0$; and *ii)* α_b is strictly increasing[12]

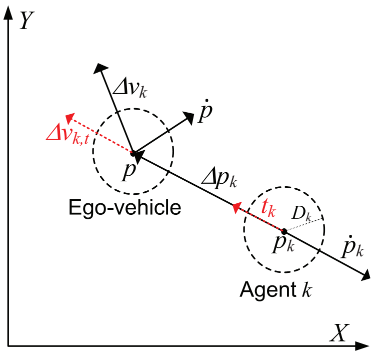


Fig. 2. Problem setup for collision avoidance.

Algorithm 2 Barrier functions for obstacle avoidance

```
for $k = 1$ to $n_o$ do
    $\mathcal{C}_{O,k}(p_k, \dot{p}_k) = \{x \in \mathcal{X} : l_{O,k}(x, p_k, \dot{p}_k) \geq 0\}$
    $l_{O,k}(x, p_k, \dot{p}_k) = \tilde{l}_{O,k}(C_p x, C_p f_x(x), p_k, \dot{p}_k)$
    $\tilde{l}_{O,k}(p, \dot{p}, p_k, \dot{p}_k) = \frac{(p - p_k)^T}{\|p - p_k\|}(\dot{p} - \dot{p}_k) + \sqrt{2\overline{\Delta a}_k(\|p - p_k\| - D_{s,k})}$
    $\mathcal{U}_{\mathcal{C}_{O,k}}(x, p_k, \dot{p}_k, \ddot{p}_k) = \{u \in \mathcal{U} : \frac{\partial l_{O,k}}{\partial x} G_x(x) u + b_k(.) \geq 0\}$
    $b_k(x, p_k, \dot{p}_k, \ddot{p}_k) = \frac{\partial l_{O,k}}{\partial x} f_x(x) + \frac{\partial l_{O,k}}{\partial p_k} \dot{p}_k + \frac{\partial l_{O,k}}{\partial \dot{p}_k} \ddot{p}_k + \underline{\varepsilon}_{d,k}(x) + \alpha_{b,k}(l_{O,k})$
end for{ $C_p$ extracts $p$ from $x$, i.e., $p = C_p x$, $l_{O,k}$ is the short notation for $l_{O,k}(x, p_k, \dot{p}_k)$ }
```

this component is positive ($\Delta v_{k,t} > 0$), then the agent k is moving away from the ego-vehicle, while negative values might lead to collisions. Thus, to prevent collisions, the tangent component needs to be non-negative. With this goal in mind, let us compute the time (T_b) and distance (D_b) needed to bring $\Delta v_{k,t}$ to zero. To compute these values, we will assume that the relative acceleration $\Delta a_k = \ddot{p} - \ddot{p}_k$ is constant

$$\overline{\Delta a}_k = \bar{a} + \sigma_k \bar{a}_k \qquad (21)$$

where $\bar{a}$ is the maximum acceleration of the ego-vehicle along t_k, $\bar{a}_k$ the maximum acceleration of agent k, and σ_k is the *degree-of-cooperation* with agent k. This latter factor allows us to incorporate additional information about the behavior of agent k - which might be uncertain - into the problem formulation. For example, if $\sigma_k = 1$, then it is assumed that both agents will try to avoid collision by applying their maximum possible acceleration. An extreme non-cooperative case is obtained with $\sigma_k = -1$, which represents the worst-case scenario where agent k is on collision course with the ego-vehicle at maximum acceleration. Based on these considerations, one can compute:

$$T_b = -\Delta v_{k,t}/\overline{\Delta a}_k, \quad D_b = -\int_0^{T_b} (\Delta v_{k,t} + \overline{\Delta a}_k t) dt = \frac{\Delta v_{k,t}^2}{(2\overline{\Delta a}_k)}$$

From a collision mitigation perspective, the relative distance between agents should fulfill the safety constraint $\|\Delta p_k\| \geq D_b + D_{s,k}$. Since $\Delta v_{k,t} = t_k^T \Delta v_k$, together with the fact that only situations with $\Delta v_{k,t} < 0$ are considered, allows us to re-write this safety constraint as:

$$-\frac{\Delta p_k^T}{\|\Delta p_k\|} \Delta v_k \leq \sqrt{2\overline{\Delta a}_k(\|\Delta p_k\| - D_{s,k})} \qquad (22)$$

By expanding the relative positions and velocities with absolute values, this safety constraint is then used to construct the *collision-free safe set* $\mathcal{C}_{O,k}$ and *collision-free input set* $\mathcal{U}_{\mathcal{C}_{O,k}}$, as detailed in Algorithm 2. Collision mitigation with all agents is achieved by ensuring that the supervisor's correction action fulfills $u_N + \Delta u \in \mathcal{U}_{\mathcal{C}_O} = \cap_{k \in \mathcal{N}_o} \mathcal{U}_{\mathcal{C}_{O,k}}$, which makes the collision-free set $\mathcal{C}_O = \cap_{k \in \mathcal{N}_o} \mathcal{C}_{O,k}$ forward invariant (see Lemma 1).

3) Final Optimization Problem: Inserting $\mathcal{U}_\mathcal{C} = \mathcal{U}_{\mathcal{C}_\mathcal{T}} \cap \mathcal{U}_{\mathcal{C}_O}$ into (18), leads to the following expanded version of the supervisor's optimization problem:

$$\min_{\Delta u, \Delta \tau, s} \quad J_s(s) + J_\tau(\Delta \tau) + J_u(u_N + \Delta u)$$

$$s.t. \quad u_N + \Delta u \in \mathcal{U} \qquad (23a)$$

$$\Delta \tau = G(x) \Delta u \qquad (23b)$$

$$2 z^T PBE(x) \Delta \tau \leq s, \quad s \geq 0 \qquad (23c)$$

$$-q_j^T C_v G_x(x)(u_N + \Delta u) \leq c_j(x), \quad j \in \mathcal{N}_T \qquad (23d)$$

$$-\frac{\partial l_{O,k}}{\partial x} G_x(x)(u_N + \Delta u) \leq b_k(.), \quad k \in \mathcal{N}_o \qquad (23e)$$

The above optimization problem is convex. This property, which can be demonstrated by exploiting the fact that both problem's constraints and cost function are convex [15], is particularly attractive because it allows us to obtain global-optimal solutions for the TTC's supervision problem.

V. Simulation Results

This section presents the validation of the supervised TTC through numerical simulations. The simulation environment is based on Matlab&Simulink, and contains the non-linear vehicle model presented in Section II, extended with vertical- and actuator dynamics. To efficiently solve the supervisor's optimization (23), which can be posed as second-order cone programming problem, we employed the *embedded conic solver* (ECOS) [16] together with cubic barrier functions ($\alpha_b(l) = kl^3$, $k > 0$). Similarly to [2], the TTC's performance is evaluated using the tangential (e_x) and normal (e_y) components of the tracking error, represented in the local vehicle coordinates, $e_{xy} = \begin{bmatrix} e_x & e_y \end{bmatrix}^T = C_e R^{-1}(\psi) e$, where $R(\psi)$ is defined in (1) and C_e is a constant[4].

In the scenario considered here, the reference trajectory is a straight line (along the X axis), generated with a constant

[4] $C_e = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}^T$

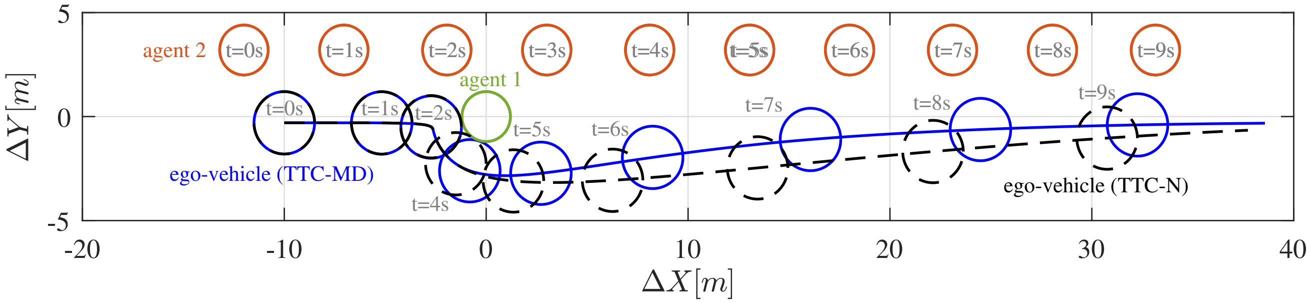
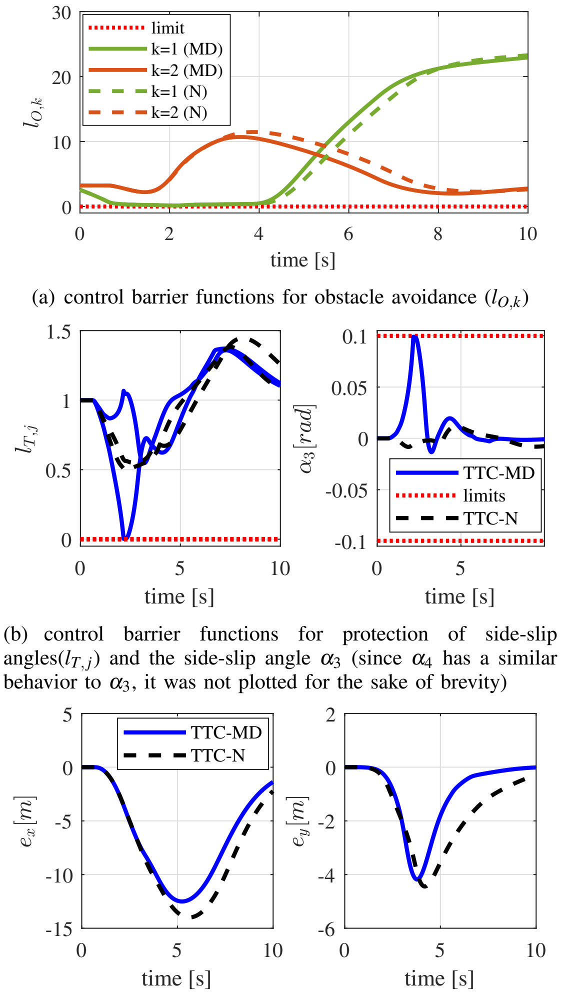


Fig. 3. Position of ego-vehicle and agent 2 represented in the coordinate frame fixed with agent 1 ($\Delta p_1 = p - p_1 = [\Delta X \; \Delta Y]^T$).



(a) control barrier functions for obstacle avoidance ($l_{O,k}$)

(b) control barrier functions for protection of side-slip angles($l_{T,j}$) and the side-slip angle α_3 (since α_4 has a similar behavior to α_3, it was not plotted for the sake of brevity)

(c) tangential (e_x) and normal (e_y) tracking errors

Fig. 4. Simulation results for the obstacle avoidance maneuver.

velocity (10 m/s). To investigate the effect of the proposed performance degradation approach, the supervised TTC is simulated with two different w_s weights. This first setting, called TTC-normal (*TTC-N*), uses $w_s = 0$, while the second employs the weight $w_s = 2$ and is referred as TTC with minimum performance degradation (*TTC-MD*). Besides the ego-vehicle, the simulative scenario also considers two non-cooperative agents ($\sigma_k = 0, k \in \{1,2\}$). Agent 1 is driving with constant velocity (5m/s), along the same lane as the

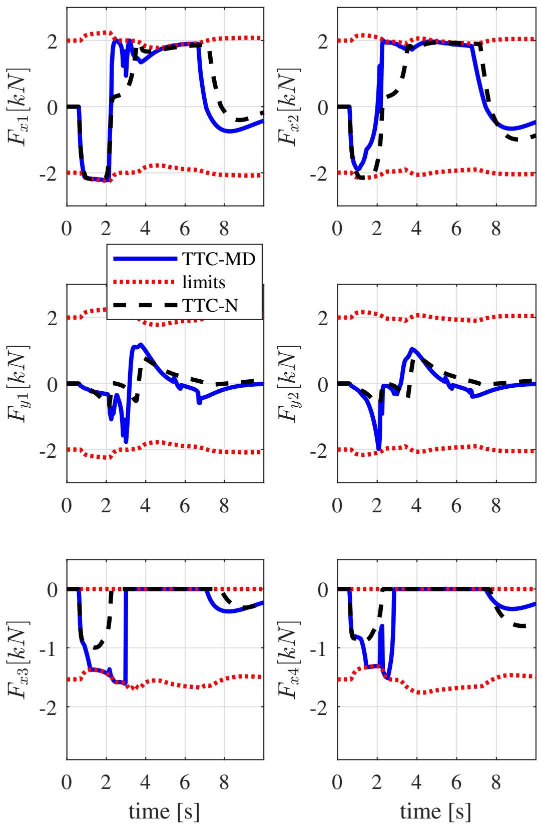


Fig. 5. Control inputs u. To simplify the time-domain representation of $\mathcal{U}$, only the worst-case $\pm\mu_i F_{zi}$ limits (dotted red lines) are shown.

ego-vehicle, having an initial longitudinal offset of 10m; agent 2 is positioned in an adjacent lane, with constant velocity of 10m/s. Additionally, the road is characterized by low adhesion conditions ($\mu_i = 0.4$). Fig. 3 depicts this simulation scenario, as well as the path of the ego-vehicle, represented in the coordinate frame fixed with the agent 1. The results reveal that both TTC-N and TTC-MD are able to avoid collision with agent 1. From 0s to 1s, the control barrier function associated with agent 1 (see $l_{O,1}$ in Fig. 4(a)) quickly approaches zero, signaling an eminent clash. This triggers aggressive braking to avoid collision – see forces F_{xi} during the period $[1,2]$s (Fig. 5) – followed by an overtaking maneuver such that both TTCs recover nominal performance and ”catch up” with the reference trajectory. The results

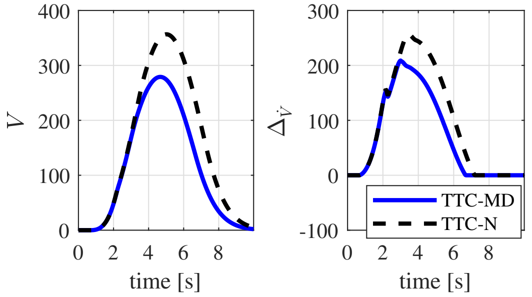


Fig. 6. Lyapunov function V and perturbation $\Delta_{\dot{V}}$ during the obstacle avoidance maneuver.

of Fig.4(c) reveal that, in comparison with the TTC-N, the TTC-MD reduces the tangential and normal tracking errors. The key reason for this superior transient behavior lies in the Lyapunov-based constraint (18c). More specifically, the increase in w_s penalizes the perturbation $\Delta_{\dot{V}}$, which reduces the value of Lyapunov function V (see Fig.6) and improves the TTC-MD response. This superior response comes, however, with a price: the collision avoidance maneuver becomes more aggressive, increasing the rear side-slip angles and bringing the vehicle closer to its driving limits. Nonetheless, thanks to the safety constraint $\mathcal{C}_{\mathcal{T}}$, the TTC-MD keeps the rear side-slip angles within a safe operating envelope (Fig.4(b)) and preserves vehicle stability.

VI. Conclusions

An optimization-based framework for supervision of trajectory tracking controllers (TTCs) was proposed in this work. This framework enhances existing TTCs with the capability to handle input and multiple safety constraints. We showed how zeroing control barrier functions (ZCBF) can be used to systematically handle safety constraints, including vehicle stability limits and collision avoidance. To minimize the performance degradation due to supervision's correction actions, the ZCBFs were combined with Lyapunov-based constraints and convex optimization methods. The supervisor's safety and minimal invasiveness properties were demonstrated through a numerical simulation example involving a collision avoidance scenario. In a future work, we will investigate how uncertainties in the environment's perception affect the design and parameterization of ZCBFs.

Appendix - Auxiliary Variables

Variables B_u and $f_y(x)$ are defined as:

$$B_u(x) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \frac{\tilde{F}_{y3}(v)}{\mu_3 F_{z3}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & \frac{\tilde{F}_{y4}(v)}{\mu_4 F_{z4}} \end{bmatrix}, \quad f_y(v) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \tilde{F}_{y3}(v) \\ 0 \\ \tilde{F}_{y4}(v) \end{bmatrix}$$

Additionally,

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned} \Gamma(x, \ddot{h}_r) &= -v_\psi \begin{bmatrix} v_y \cos(\psi) + v_x \sin(\psi) + L_x v_\psi \cos(\psi) \\ v_y \sin(\psi) - v_x \cos(\psi) + L_x v_\psi \sin(\psi) \\ 0 \end{bmatrix} - \ddot{h}_r \\ &\quad + E(x) C_v f_x(x) \end{aligned} \tag{24}$$

$$E(x) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & -L_x \sin(\psi) \\ \sin(\psi) & \cos(\psi) & L_x \cos(\psi) \\ 0 & 0 & 1 \end{bmatrix} \tag{25}$$

where C_v extracts the v component from x, i.e. $v = C_v x$. The actuation cost J_u penalizes the tires' friction usage [10]:

$$J_u(u) = \sum_{i \in \{1,2\}} \frac{F_{xi}^2 + F_{yi}^2}{(\mu_i F_{zi})^2} + \sum_{i \in \{3,4\}} \frac{F_{xi}^2 + \left(\tilde{F}_{yi}(x)^2 (1 - \frac{F_{xi}}{\mu_i F_{zi}})\right)}{(\mu_i F_{zi})^2}$$

References

[1] B. Paden, M. Cap, S. Z. Yong, D. Yershov, and E. Frazzoli, "A Survey of Motion Planning and Control Techniques for Self-driving Urban Vehicles," *IEEE Transactions on Intelligent Vehicles*, 2016.

[2] D. Calzolari, B. Schrmann, and M. Althoff, "Comparison of Trajectory Tracking Controllers for Autonomous Vehicles," in *20th IEEE International Conference on Intelligent Transportation Systems*, 2017.

[3] S. Dominguez, A. Ali, G. Garcia, and P. Martinet, "Comparison of lateral controllers for autonomous vehicle: Experimental results," in *19th Int. Conf. on Intelligent Transportation Systems*, 2016.

[4] H. Park and J. C. Gerdes, "Analysis of Feasible Tire Force Regions for Optimal Tire Force Allocation with Limited Actuation," *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 3, 2017.

[5] G. Tagne, R. Talj, and A. Charara, "Design and Comparison of Robust Nonlinear Controllers for the Lateral Dynamics of Intelligent Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 3, pp. 796–809, Mar. 2016.

[6] S. C. Peters, E. Frazzoli, and K. Iagnemma, "Differential flatness of a front-steered vehicle with tire force control," in *IEEE International Conference on Intelligent Robots and Systems*, 2011.

[7] D. Hess, M. Althoff, and T. Sattel, "Comparison of trajectory tracking controllers for emergency situations," in *Intelligent Vehicles Symposium (IV)*, 2013.

[8] V. Turri, A. Carvalho, H. E. Tseng, K. H. Johansson, and F. Borrelli, "Linear model predictive control for lane keeping and obstacle avoidance on low curvature roads," in *16th International IEEE Conference on Intelligent Transportation Systems*, Oct. 2013.

[9] H. B. Pacejka, *Tyre and vehicle dynamics*. Butterworth-Heinemann, 2002.

[10] R. de Castro and J. Brembeck, "Lyapunov-based Control Allocation for Over-actuated Nonlinear Systems," *Manuscript in preparation*, 2018.

[11] H. Khalil, *Nonlinear Control*, 3rd ed. Prentice Hall, 2002.

[12] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control Barrier Function Based Quadratic Programs for Safety Critical Systems," *IEEE Transactions on Automatic Control*, 2017.

[13] C. E. Beal and J. C. Gerdes, "Model Predictive Control for Vehicle Stabilization at the Limits of Handling," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1258–1269, Jul. 2013.

[14] L. Wang, A. D. Ames, and M. Egerstedt, "Safety Barrier Certificates for Collisions-Free Multirobot Systems," *IEEE Transactions on Robotics*, 2017.

[15] S. Boyd and L. Vandenberghe, *Convex optimization*, 18th ed. Cambridge: Cambridge Univ. Press, 2015.

[16] A. Domahidi, E. Chu, and S. Boyd, "ECOS: An SOCP solver for embedded systems," in *European Control Conference*, 2013.