MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

Friction-Adaptive Stochastic Predictive Control for Trajectory Tracking of Autonomous Vehicles

Vaskov, Sean; Quirynen, Rien; Menner, Marcel; Berntorp, Karl

TR2022-065 June 11, 2022

Abstract

This paper addresses the trajectory-tracking prob- lem under uncertain road-surface conditions for autonomous vehicles. We develop a stochastic nonlinear model-predictive controller (SNMPC) that learns the tire–road friction rela- tionship online using standard automotive-grade sensors. We learn the tire-friction function using a Bayesian approach, where the friction curve is modeled as a Gaussian process. The estimator outputs the estimate of the tire-friction model as well as the uncertainty function of the estimate, which expresses the confidence in the model for different driving regimes. The SNMPC exploits the uncertainty estimate in its prediction model to take proper action. We validate the approach using the high-fidelity vehicle simulator CarSim and compare against various nominal NMPC approaches. The results indicate more than six times better performance for the proposed adaptive SNMPC in closed-loop cost over the simulation time

American Control Conference (ACC) 2022

© 2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

Friction-Adaptive Stochastic Predictive Control for Trajectory Tracking of Autonomous Vehicles

Sean Vaskov[1,2], Rien Quirynen[1], Marcel Menner[1], and Karl Berntorp[1]

***Abstract*— This paper addresses the trajectory-tracking problem under uncertain road-surface conditions for autonomous vehicles. We develop a stochastic nonlinear model-predictive controller (SNMPC) that learns the tire–road friction relationship online using standard automotive-grade sensors. We learn the tire-friction function using a Bayesian approach, where the friction curve is modeled as a Gaussian process. The estimator outputs the estimate of the tire-friction model as well as the uncertainty function of the estimate, which expresses the confidence in the model for different driving regimes. The SNMPC exploits the uncertainty estimate in its prediction model to take proper action. We validate the approach using the high-fidelity vehicle simulator CarSim and compare against various nominal NMPC approaches. The results indicate more than six times better performance for the proposed adaptive SNMPC in closed-loop cost over the simulation time.**

I. Introduction

The strong push recently in the automotive industry for introducing new technologies related to automated driving (AD) and advanced driving-assistance systems (ADAS), has led to predictive information being readily available to the vehicle controllers [1]. Model-predictive control (MPC) is suitable for vehicle control as it naturally integrates predictive information in its problem formulation [2], and MPC has successfully been applied to vehicle trajectory tracking and stability control [3], [4]. The key benefit of MPC is its ability to incorporate constraint handling as well as prediction models in its optimal control problem (OCP). However, because MPC relies on a model, the model needs to represent the actual vehicle behavior sufficiently well. Specifically, since the main actuation of the vehicle is done by altering the forces generated by the tire–road contact, it is imperative to have a sensible guess about the road surface the car operates on, as otherwise safe operation of the vehicle can be endangered and performance can be limited [5], [6].

In this paper, we develop an MPC that adapts in real time to different road-surface conditions by using sensors that are standard in production vehicles. The interaction between tire and road is highly nonlinear, and the function describing the nonlinear relation varies heavily based on the road surface and other tire properties [7]. When driving close to the adhesion limits, which may happen in emergency maneuvers, on unpaved roads, or on wet and icy roads, the nonlinear part of the tire-force function may be excited, and the full tire curve shape must be considered.

[1]Mitsubishi Electric Research Laboratories (MERL), 02139 Cambridge, MA, USA. Email: karl.o.berntorp@ieee.org

[2]University of Michigan, 40819, Ann Arbor, MI, USA

Obtaining data for the nonlinear part of the tire-friction function is challenging, and driving in the nonlinear region of the tire-friction function before a reliable model of the same is obtained possibly dangerous. Another difficulty when learning the tire-friction function using automotive-grade sensors is that the amount of sensors is limited, and they are relatively low grade. Moreover, the sensors only provide indirect measurements of the friction and the vehicle state.

To resolve these issues, in this paper, we couple an online tire-friction estimator with a stochastic nonlinear MPC (SNMPC). The estimator is fully Bayesian and models the tire-friction curve as a Gaussian process (GP) with unknown and time-varying mean and covariance function, leading to a GP state-space model (GP-SSM). In particular, we leverage a recently proposed method for real-time joint state estimation and learning of the state-transition function [8], where a computationally efficient formulation of GP-SSMs is combined with particle filtering [9] for jointly estimating online the state and associated state-transition function. By using the high-fidelity vehicle dynamics simulator CarSim [10], we show that the estimator: (i), behaves in a manner suitable for safety-critical vehicle control while only demanding standard sensors and (ii), predicts larger uncertainty in the regions of the function that have not been sufficiently excited. Friction-adaptive NMPC has been explored before, for example, in [11], [12]. However, unlike previous work, we explicitly leverage the uncertainty estimates by coupling the estimator with an SNMPC, which can utilize the Bayesian estimator.

The SNMPC formulation considers individual chance constraints approximated using online linearization-based covariance propagation [13], given the uncertainty information from the estimator. The resulting approximate stochastic nonlinear OCP is solved using an efficient block-sparse quadratic programming (QP) solver [14] and a tailored Jacobian approximation within the adjoint-based sequential QP (SQP) method [15]. We have reported on methods for friction-adaptive MPC [4], [16]. This work differs in that [4] chooses from a fixed library of pre-determined tire-friction functions and does not utilize SNMPC. While [16] uses SNMPC, it employs a friction estimator targeting the linear region of the tire-friction curve [17].

Notation: The notation $\mathcal{R}(a)$ means the 2D rotation matrix of an angle a. Vectors are shown in bold, $\boldsymbol{x}$, we denote the stacking of two vectors $\boldsymbol{a}$, $\boldsymbol{b}$ by $[\boldsymbol{a}, \boldsymbol{b}]$, and constraints between vectors are intended componentwise. Throughout, $\boldsymbol{x} \sim \mathcal{N}(\hat{\boldsymbol{x}}, \boldsymbol{\Sigma})$ indicates that $\boldsymbol{x} \in \mathbb{R}^{n_x}$ is Gaussian distributed with mean $\hat{\boldsymbol{x}}$ and covariance $\operatorname{cov}(\boldsymbol{x}) = \boldsymbol{\Sigma}$. Matrices are indicated in capital bold font as $\boldsymbol{X}$. With $p(\boldsymbol{x}_{0:k}|\boldsymbol{y}_{0:k})$, we mean the posterior density function of the state trajectory $\boldsymbol{x}_{0:k}$ from time step 0 to time step k given the measurement sequence $\boldsymbol{y}_{0:k} := \{\boldsymbol{y}_0, \ldots, \boldsymbol{y}_k\}$, and $\boldsymbol{x}^i_{0:k}$ is the i^{th} realization of $\boldsymbol{x}_{0:k}$. The notation $\boldsymbol{f}(\boldsymbol{x}) \sim \mathcal{GP}(\hat{\boldsymbol{f}}(\boldsymbol{x}), \boldsymbol{\Sigma}(\boldsymbol{x}))$ means that the function $\boldsymbol{f}(\boldsymbol{x})$ is a realization from a GP with a mean function $\hat{\boldsymbol{f}}(\cdot)$ and covariance $\boldsymbol{\Sigma}(\cdot)$. For a continuous-time signal $\boldsymbol{x}(t)$ sampled with period T_s, x_k denotes the k^{th} sample, that is, $\boldsymbol{x}_k = \boldsymbol{x}(kT_s)$. The norm of the quadratic form of a vector $\boldsymbol{x}$ and matrix Q is given by $\|\boldsymbol{x}\|^2_Q = \boldsymbol{x}^\top Q \boldsymbol{x}$.

II. Modeling and Problem Formulation

We consider the single-track vehicle model, where the left and right track of the car are lumped into a single centered track. Roll and pitch dynamics are ignored, resulting in two translational and one rotational degrees of freedom. The single-track model is sufficient in most evasive maneuvers, because the focus of such maneuvers is on preserving safety rather than achieving optimality [5], [6]. Additionally, the single-track model results in a reduced computational load, which is desirable in automotive applications [2]. With the longitudinal and lateral velocities in the vehicle frame, v^X, v^Y, and the yaw rate, $\dot{\psi}$, as states, the single-track model is

$$\dot{v}^X - v^Y\dot{\psi} = \frac{1}{m}(F^x_f \cos(\delta_f) + F^x_r - F^y_f \sin(\delta_f)), \tag{1a}$$

$$\dot{v}^Y + v^X\dot{\psi} = \frac{1}{m}(F^y_f \cos(\delta_f) + F^y_r + F^x_f \sin(\delta_f)), \tag{1b}$$

$$I_{zz}\ddot{\psi} = l_f F^y_f \cos(\delta_f) - l_r F^y_r + l_f F^x_f \sin(\delta_f), \tag{1c}$$

where $F^x_i = F^z_i \mu^x_i(\sigma_i)$, $F^y_i = F^z_i \mu^y_i(\alpha_i)$ are the total longitudinal/lateral forces in the tire frame for the lumped left and right tires where the friction functions $\mu^x_i(\cdot)$, $\mu^y_i(\cdot)$ are provided by the estimator, σ_i is the longitudinal wheel slip, α_i is the slip angle, and subscripts $i \in \{f, r\}$ denote front and rear, respectively. The normal forces resting on the lumped front/rear wheels F^z_i, $i \in \{f, r\}$, are $F^z_f = m\, g\, l_r/l$, $F^z_r = m\, g\, l_f/l$, where g is the gravity acceleration, l_f, l_r are the distances of front and rear axles from the center of gravity, and $l = l_f + l_r$ is the vehicle wheel base. Also, m is the vehicle mass, I_{zz} is the vehicle inertia about the vertical axis, and δ_f is the front wheel (road) steering angle. The global vehicle position $\boldsymbol{p} = (p^{\mathbf{X}}, p^{\mathbf{Y}})$ is obtained from

$$\begin{bmatrix}\dot{p}^{\mathbf{X}} & \dot{p}^{\mathbf{Y}}\end{bmatrix}^\top = \mathcal{R}(\psi)\begin{bmatrix}v^X & v^Y\end{bmatrix}^\top. \tag{2}$$

The slip angles α_i and slip ratios σ_i are defined as in [18],

$$\alpha_i = -\arctan\left(v^y_i / v^x_i\right), \quad \sigma_i = \frac{r_w\omega_i - v^x_i}{\max(r_w\omega_i, v^x_i)}, \tag{3}$$

where r_w is the wheel radius, and v^x_i and v^y_i are the longitudinal and lateral wheel velocities for wheel i.

A. Problem Formulation

The objective is to design a friction-adaptive control strategy that makes the vehicle motion follow a time-dependent reference trajectory $(p^{\mathbf{X}}_{\text{ref}}(\cdot), p^{\mathbf{Y}}_{\text{ref}}(\cdot), \psi_{\text{ref}}(\cdot), v^X_{\text{ref}}(\cdot))$, possibly generated in real time with an adequate preview, while operating over different surfaces and environmental conditions. The control strategy needs to estimate the vehicle state and the tire-friction model with the controller activated. In addition, since the estimation uncertainty may vary over time and it may vary in different regions of the tire-friction curve, the proposed SNMPC directly takes into account this time- and state-dependent uncertainty information in real time.

III. Stochastic Model-Predictive Control for Real-Time Vehicle Control

Based on (1)–(3) and using a time discretization with sampling period T_s, the complete vehicle model is

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\mu}_k), \tag{4}$$

where $\boldsymbol{x}_k \in \mathbb{R}^{n_x}$ is the state, $\boldsymbol{u}_k \in \mathbb{R}^{n_u}$ the control input, and $\boldsymbol{\mu}_k(\boldsymbol{x}, \boldsymbol{u})$ is the unknown tire-friction function. The tire-friction function is modeled as a GP with time-varying mean function $\hat{\boldsymbol{\mu}}_k(\boldsymbol{x}, \boldsymbol{u})$ and covariance function $\boldsymbol{\Sigma}_k(\boldsymbol{x}, \boldsymbol{u})$ according to $\boldsymbol{\mu}_k(\boldsymbol{x}, \boldsymbol{u}) \sim \mathcal{GP}(\hat{\boldsymbol{\mu}}_k(\boldsymbol{x}, \boldsymbol{u}), \boldsymbol{\Sigma}_k(\boldsymbol{x}, \boldsymbol{u}))$, where the mean and covariance are provided online by the estimator. The state and control vector in the SNMPC formulation are

$$\begin{aligned}\boldsymbol{x} &:= \begin{bmatrix}p^{\mathbf{X}},\ p^{\mathbf{Y}},\ \psi,\ v^X,\ v^Y,\ \dot{\psi},\ \delta_f\end{bmatrix}^\top, \\ \boldsymbol{u} &:= \begin{bmatrix}\dot{\delta}_f,\ \omega_f,\ \omega_r\end{bmatrix}^\top.\end{aligned} \tag{5}$$

At each sampling time, based on the state estimate $\hat{\boldsymbol{x}}_t$ and covariance $\boldsymbol{P}_t$ at the current time step t, the SNMPC solves

$$\min_{\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{P}_k} \sum_{k=0}^{N-1} l_k(\boldsymbol{x}_k, \boldsymbol{u}_k) + l_N(\boldsymbol{x}_N) \tag{6a}$$

$$\text{s.t.} \quad \boldsymbol{x}_0 = \hat{\boldsymbol{x}}_t, \quad \boldsymbol{P}_0 = \boldsymbol{P}_t, \tag{6b}$$

$$\forall k \in \{0, \ldots, N-1\}:$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k, \hat{\boldsymbol{\mu}}_t(\boldsymbol{x}_k, \boldsymbol{u}_k)), \tag{6c}$$

$$\boldsymbol{P}_{k+1} = \boldsymbol{F}_k \boldsymbol{P}_k \boldsymbol{F}_k^\top + \boldsymbol{G}_k \boldsymbol{\Sigma}_t(\boldsymbol{x}_k, \boldsymbol{u}_k) \boldsymbol{G}_k^\top, \tag{6d}$$

$$Pr\left(\mathbf{c}(\boldsymbol{x}_k, \boldsymbol{u}_k) \le 0\right) \ge 1 - \epsilon, \tag{6e}$$

where the overall control action is in the feedforward-feedback form $\boldsymbol{u}_k = \boldsymbol{u}_{\text{ref},k} + K(\boldsymbol{x}_k - \boldsymbol{x}_{\text{ref},k}) + \Delta\boldsymbol{u}_k$ due to a prestabilizing controller, and $\boldsymbol{F}_k$, $\boldsymbol{G}_k$ are the Jacobian matrices of (4) linearized with respect to $\boldsymbol{x}$ and $\boldsymbol{\mu}$, respectively. The state covariance propagation in (6d) corresponds to the extended Kalman filter (EKF).

In linearization-based SNMPC, the model disturbance is Gaussian-assumed and state-independent [13], [15], whereas the tire-friction function is a GP and state-dependent. Hence, we need to adapt the covariance propagation (6d) to account for this state dependence by integrating the estimated tire-friction function with the SNMPC problem (6). This amounts to modifying the calculation of the Jacobian matrices $\boldsymbol{F}_k$ and $\boldsymbol{G}_k$, as described further in Section V.

A. Objective Function and Inequality Constraints

In our OCP, we consider the stage cost and terminal cost in (6a) to be the least-squares functions

$$\begin{aligned}l_k(\cdot) &= \frac{1}{2}\|\boldsymbol{x}_k - \boldsymbol{x}_{\text{ref},k}\|^2_Q + \frac{1}{2}\|\boldsymbol{u}_k - \boldsymbol{u}_{\text{ref},k}\|^2_R + r_s s_k, \\ l_N(\cdot) &= \frac{1}{2}\|\boldsymbol{x}_N - \boldsymbol{x}_{\text{ref},N}\|^2_{Q_N},\end{aligned} \tag{7}$$

which includes a term for both state and control reference tracking, and an L1 term for penalizing the slack variable $s_k \geq 0$. The constraints $\mathbf{c}(\boldsymbol{x}_k, \boldsymbol{u}_k) \leq 0$ in the OCP (6) consist of geometric and physical limitations on the system. In practice, it is important to reformulate these requirements as soft constraints, based on the slack variable s_k, since otherwise the problem may become infeasible due to unknown disturbances and modeling errors. The constraints in (6e) therefore include soft bounds on

$$p^{\mathbf{Y}}_{\min} - s_k \leq p^{\mathbf{Y}}_k \leq p^{\mathbf{Y}}_{\max} + s_k, \tag{8a}$$

$$-\delta_{f,\max} - s_k \leq \delta_{f,k} \leq \delta_{f,\max} + s_k, \tag{8b}$$

$$-\dot{\delta}_{f,\max} - s_k \leq \dot{\delta}_{f,k} \leq \dot{\delta}_{f,\max} + s_k, \tag{8c}$$

$$\omega_{\min} - s_k \leq \omega_{i,k} \leq \omega_{\max} + s_k, \; i \in \{f, r\}, \tag{8d}$$

$$-\alpha_{\max} - s_k \leq \alpha_{i,k} \leq \alpha_{\max} + s_k, \; i \in \{f, r\}. \tag{8e}$$

B. Probabilistic Chance Constraints

To enforce the probabilistic chance constraints in (6e), we approximate them as deterministic constraints as in [13], where the j^{th} constraint is written as

$$c_j(\boldsymbol{x}_k, \boldsymbol{u}_k) + \nu \sqrt{\frac{\partial c_j}{\partial \boldsymbol{x}_k} \boldsymbol{P}_k \frac{\partial c_j}{\partial \boldsymbol{x}_k}^T} \leq 0, \tag{9}$$

where ν is referred to as the back-off coefficient and depends on the desired probability threshold ϵ and assumptions about the resulting state distribution. The back-off coefficient for Cantelli's inequality, $\nu = \sqrt{\frac{1-\epsilon}{\epsilon}}$, holds regardless of the underlying distribution but is conservative. In this work, we assume normal-distributed state trajectories and set

$$\nu = \sqrt{2}\, \text{erf}^{-1}(1 - 2\epsilon), \tag{10}$$

where $\text{erf}^{-1}(\cdot)$ is the inverse error function.

C. Online SNMPC and Software Implementation

We solve the OCP in (6) at each time step of control by a novel variant of the real-time iteration (RTI) algorithm [19], using the adjoint-based SQP method that was proposed in [15]. The algorithm uses a tailored Jacobian approximation that allows for the numerical elimination of the covariance matrices from the SQP subproblem, which reduces the computation time considerably and allows for real-time feasible implementations of stochastic nonlinear MPC as illustrated in [15]. The real-time algorithm performs one adjoint-based SQP iteration per time step of control, and uses a continuation-based warm starting of the state and control trajectories from one time step to the next.

The resulting SNMPC implementation uses the embedded QP solver `PRESAS` [14], which applies block-structured factorization techniques with low-rank updates to preconditioning of an iterative solver within a primal active-set algorithm for MPC applications. The nonlinear function and derivative evaluations, for the preparation of each SQP subproblem, are performed using algorithmic differentiation (AD) and C code generation in `CasADi` [20]. In addition, a standard line search method is used [21] to improve the closed-loop convergence of the SQP-based SNMPC controller.

IV. Bayesian Tire-Friction Learning by Gaussian-Process State-Space Models

The estimator is targeted for embedded automotive-grade hardware and sensors. We use a recently developed method for *jointly* estimating the tire-friction function $\boldsymbol{\mu}$ and the vehicle state $\boldsymbol{x}$ only using sensors available in production cars, namely wheel-speed sensors and inexpensive accelerometers and gyroscopes. We briefly outline the formulation of the method and refer to [8], [22] for a more complete description.

Remark 1: We focus on the estimation of the lateral tire friction ($n_\mu = 2$). The extension to the longitudinal case is analogous. We focus on the lateral vehicle dynamics, because usually these are the most critical for vehicle control.

A. Estimation Model

The method uses the single-track model (1) and models the tire-friction components as static functions of the slip quantities, which in the lateral case results in

$$\mu_i^y = f_i^y(\alpha_i(\boldsymbol{x}^{\text{e}}, \boldsymbol{u}^{\text{e}})), \quad i \in \{f, r\}, \tag{11}$$

where the state and input vector for the estimator are $\boldsymbol{x}^{\text{e}} = [v^Y, \dot{\psi}]^\top$, $\boldsymbol{u}^{\text{e}} = [\delta, v^X]^\top$. We let $\boldsymbol{\alpha} = [\alpha_f, \alpha_r]^\top$ and model the friction vector as a realization from a GP with mean function $\hat{\boldsymbol{\mu}}$ and covariance function $\boldsymbol{\Sigma}$,

$$\boldsymbol{\mu}^y(\boldsymbol{\alpha}(\boldsymbol{x}^{\text{e}}, \boldsymbol{u}^{\text{e}})) \sim \mathcal{GP}(\hat{\boldsymbol{\mu}}(\boldsymbol{\alpha}(\boldsymbol{x}^{\text{e}}, \boldsymbol{u}^{\text{e}})), \boldsymbol{\Sigma}(\boldsymbol{\alpha}(\boldsymbol{x}^{\text{e}}, \boldsymbol{u}^{\text{e}}))). \tag{12}$$

The resulting vehicle SSM is a GP-SSM where the tire friction is a GP. A bottleneck in some of the proposed GP-SSM methods is the computational load. In this paper, we use a computationally efficient reduced-rank GP-SSM framework, where the GP is approximated by a basis-function expansion using the Laplace operator eigenvalues and eigenfunctions

$$\phi^j(\alpha) = \frac{1}{\sqrt{L}} \sin\left(\frac{\pi j(\alpha + L)}{2L}\right), \quad \lambda_j = \left(\frac{\pi j}{2L}\right)^2, \tag{13}$$

defined on the interval $[-L, L]$, such that

$$\mu_i^y \approx \sum_j \gamma_i^j \, \phi^j(\alpha_i), \tag{14}$$

where the weights γ_i^j are Gaussian random variables with unknown mean and covariance, whose prior depends on the spectral density that is a function of the eigenvalues in (13). We express the prior on the coefficients γ_i^j at time step $k = 0$ as a zero-mean matrix-normal ($\mathcal{MN}$) distribution over $\boldsymbol{A}$,

$$\boldsymbol{A} \sim \mathcal{MN}(\mathbf{0}, \boldsymbol{Q}, \boldsymbol{V}), \tag{15}$$

with right covariance $\boldsymbol{Q}$ and left covariance $\boldsymbol{V}$. We write the basis-function expansion on matrix form as

$$\boldsymbol{\mu} = \underbrace{\begin{bmatrix} \gamma_f^1 & \cdots & \gamma_f^m & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \gamma_r^1 & \cdots & \gamma_r^m \end{bmatrix}}_{\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_f & \mathbf{0} \\ \mathbf{0} & \boldsymbol{A}_r \end{bmatrix}} \underbrace{\begin{bmatrix} \phi^1(\alpha_f) \\ \vdots \\ \phi^m(\alpha_f) \\ \phi^1(\alpha_r) \\ \vdots \\ \phi^m(\alpha_r) \end{bmatrix}}_{\boldsymbol{\varphi}(\boldsymbol{\alpha})}, \tag{16}$$

where γ_i^j are the weights to be learned and m is the total number of basis functions.

With (16), the vehicle dynamics estimation model is

$$\boldsymbol{x}_{k+1}^{\mathrm{e}} = \boldsymbol{f}^e(\boldsymbol{x}_k^{\mathrm{e}}, \boldsymbol{u}_k^{\mathrm{e}}) + \boldsymbol{g}^e(\boldsymbol{x}_k^{\mathrm{e}}, \boldsymbol{u}_k^{\mathrm{e}})\boldsymbol{A}\boldsymbol{\varphi}(\boldsymbol{\alpha}(\boldsymbol{x}_k^{\mathrm{e}}, \boldsymbol{u}_k^{\mathrm{e}})). \quad (17)$$

Hence, the original problem of learning the infinite-dimensional friction function $\boldsymbol{\mu}$ has been transformed to learning the matrix $\boldsymbol{A}$ in (17), which is substantially easier to do in an online setting. The lateral velocity and yaw rate are estimated as part of the state according to (1). The longitudinal velocity can be determined from the wheel speeds $\{\omega_i\}_{i=f,r}$ and the longitudinal acceleration measurement a^X.

Our measurement model is based on a setup commonly available in production cars, namely the acceleration a^Y, and yaw-rate $\dot{\psi}$, forming the measurement vector $\boldsymbol{y} = [a^Y, \dot{\psi}]^\top$. To relate $\boldsymbol{y}_k$ to the estimator state $\boldsymbol{x}_k^{\mathrm{e}}$ at each time step k, note that a^Y can be extracted from the right-hand side of (1b), and the yaw rate is a state. The measurement noise $\boldsymbol{e}_k$ is zero-mean Gaussian distributed with covariance $\boldsymbol{R}$ according to $\boldsymbol{e}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R})$. This results in the measurement model

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k^{\mathrm{e}}, \boldsymbol{u}_k^{\mathrm{e}}) + \boldsymbol{D}(\boldsymbol{x}_k^{\mathrm{e}}, \boldsymbol{u}_k^{\mathrm{e}})\boldsymbol{\mu}(\boldsymbol{\alpha}_k(\boldsymbol{x}_k^{\mathrm{e}}, \boldsymbol{u}_k^{\mathrm{e}})) + \boldsymbol{e}_k. \quad (18)$$

The measurement covariance $\boldsymbol{R}$ is assumed known a priori. This is reasonable, since the measurement noise can oftentimes be determined from prior experiments and data sheets.

Remark 2: According to well-established tire models (e.g., [23]), the tire-friction function is antisymmetric. The basis-function expansion (14) is a weighted sum of sinusoids, and we can trivially enforce antisymmetry by restricting the sum in (14) to even values for j (i.e., $j = 2, 4, \ldots, m$). This reduces the number of parameters to estimate and ensures that the estimate (11) passes through the origin.

B. Joint State and Friction-Function Learning

The vehicle state needs to be estimated concurrently with the friction function. To properly account for the inherent uncertainty and to be able to integrate properly with SNMPC, we approach the estimation problem in a Bayesian framework. Due to the nonlinearities of the tire-friction function and the vehicle model, the estimation problem is highly non-Gaussian. We therefore leverage a particle-filter (PF) based estimator that estimates the vehicle state $\boldsymbol{x}^{\mathrm{e}}$ and basis-function weights $\boldsymbol{A}$ [22]. The PF approximates the joint posterior density at each time step k as

$$p(\boldsymbol{A}_k, \boldsymbol{x}_{0:k}^{\mathrm{e}}|\boldsymbol{y}_{0:k}) = p(\boldsymbol{A}_k|\boldsymbol{x}_{0:k}^{\mathrm{e}}, \boldsymbol{y}_{0:k})\, p(\boldsymbol{x}_{0:k}^{\mathrm{e}}|\boldsymbol{y}_{0:k}). \quad (19)$$

The two densities on the right-hand side of (19) can be estimated recursively. We estimate the state trajectory density $p(\boldsymbol{x}_{0:k}^{\mathrm{e}}|\boldsymbol{y}_{0:k})$ by a set of N weighted state trajectories as

$$p(\boldsymbol{x}_{0:k}^{\mathrm{e}}|\boldsymbol{y}_{0:k}) \approx \sum_{i=1}^{N} q_k^i\, \delta_{\boldsymbol{x}_{0:k}^{\mathrm{e},i}}(\boldsymbol{x}_{0:k}^{\mathrm{e}}), \quad (20)$$

where q_k^i is the weight of the i^{th} state trajectory $\boldsymbol{x}_{0:k}^{\mathrm{e},i}$ and $\delta(\cdot)$ is the Dirac delta mass. Given the state trajectory, we can compute the sufficient statistics necessary to approximate $p(\boldsymbol{A}_k|\boldsymbol{x}_{0:k}^{\mathrm{e},i}, \boldsymbol{y}_{0:k})$ for each particle. Because we determine the state trajectory from the PF, the computations leading up to the estimation of $p(\boldsymbol{A}_k|\boldsymbol{x}_{0:k}^{\mathrm{e},i}, \boldsymbol{y}_{0:k})$ are analytic.

To determine the covariance and mean function, we marginalize out the state trajectory from $p(\boldsymbol{A}_k|\boldsymbol{x}_{0:k}^{\mathrm{e},i}, \boldsymbol{y}_{0:k})$,

$$\begin{aligned} p(\boldsymbol{A}_k|\boldsymbol{y}_{0:k}) &= \int p(\boldsymbol{A}_k|\boldsymbol{x}_{0:k}^{\mathrm{e}}, \boldsymbol{y}_{0:k}) p(\boldsymbol{x}_{0:k}^{\mathrm{e}}|\boldsymbol{y}_{0:k})\, \mathrm{d}\boldsymbol{x}_{0:k}^{\mathrm{e}} \\ &\approx \sum_{i=1}^{N} q_k^i\, p(\boldsymbol{A}_k|\boldsymbol{x}_{0:k}^{\mathrm{e},i}, \boldsymbol{y}_{0:k}), \quad (21) \end{aligned}$$

from where the mean function $\hat{\boldsymbol{A}}_k \boldsymbol{\varphi}(\boldsymbol{\alpha}_k(\boldsymbol{x}_k^{\mathrm{e}}, \boldsymbol{u}_k^{\mathrm{e}}))$ and covariance function $\boldsymbol{\Sigma}_k(\boldsymbol{\alpha}_k(\boldsymbol{x}_k^{\mathrm{e}}, \boldsymbol{u}_k^{\mathrm{e}}))$ needed in the SNMPC problem (6) can be extracted, as discussed next.

V. Online Adaptation of SNMPC to Tire-Friction Learning

In determining the mean and covariance of the tire-friction, we use that each particle retains its own estimate $\boldsymbol{A}_k^i$ together with the weight q_k^i. This results in

$$\begin{aligned} \hat{\boldsymbol{\mu}}_k &= \sum_{i=1}^{N} q_k^i \left[\hat{\boldsymbol{A}}_f^i \boldsymbol{\varphi}(\alpha_{f,k}), \hat{\boldsymbol{A}}_r^i \boldsymbol{\varphi}(\alpha_{r,k})\right]^\top, \\ \boldsymbol{\Sigma}_k &= \sum_{i=1}^{N} q_k^i \begin{bmatrix} \boldsymbol{\varphi}(\alpha_{f,k})^\top \boldsymbol{V}_{f,k}^i \boldsymbol{\varphi}(\alpha_{f,k}) Q_f & 0 \\ 0 & \boldsymbol{\varphi}(\alpha_{r,k})^\top \boldsymbol{V}_{r,k}^i \boldsymbol{\varphi}(\alpha_{r,k}) Q_r \end{bmatrix}, \end{aligned} \quad (22)$$

where $\boldsymbol{V}_{f,k}^i$, $\boldsymbol{V}_{r,k}^i$ are the posterior left covariances [8].

A. Integrating the Tire-Friction Function into SNMPC

Using the single-track model and the basis-function expansion of the friction, $\boldsymbol{\mu} \approx \boldsymbol{A}\boldsymbol{\varphi}(\boldsymbol{\alpha}(\boldsymbol{x}))$, we can write the vehicle model (4) used in the SNMPC similar to (17),

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k) + \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k)\hat{\boldsymbol{\mu}}_k(\boldsymbol{\alpha}(\boldsymbol{x}_k)). \quad (23)$$

In linearization-based SNMPC [15], the parametric uncertainty is typically modeled according to a Gaussian random variable, $\boldsymbol{\mu} \sim \mathcal{N}(\hat{\boldsymbol{\mu}}, \boldsymbol{\Sigma})$, where the uncertainty $\boldsymbol{\mu}$ is state-independent. This leads to Jacobian matrices $\boldsymbol{F}_k = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\boldsymbol{x}_k, \boldsymbol{u}_k, \hat{\boldsymbol{\mu}}_k)$ and $\boldsymbol{G}_k = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{w}}(\boldsymbol{x}_k, \boldsymbol{u}_k, \hat{\boldsymbol{\mu}}_k)$ in (6d). However, we have that the friction uncertainty is a function modeled according to a GP at each time step k, $\boldsymbol{\mu}(\boldsymbol{\alpha}(\boldsymbol{x})) \sim \mathcal{GP}\left(\hat{\boldsymbol{\mu}}_k(\boldsymbol{\alpha}(\boldsymbol{x})), \boldsymbol{\Sigma}(\boldsymbol{\alpha}(\boldsymbol{x}))\right)$. Hence, we want to determine the equivalent to (6d) for the state-dependent uncertainty $\boldsymbol{\mu}(\boldsymbol{\alpha}(\boldsymbol{x}))$, by finding expressions for the involved Jacobians $\boldsymbol{F}_k$, $\boldsymbol{G}_k$. From (23), by using the chain rule,

$$\begin{aligned} \boldsymbol{F}_k = &\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\boldsymbol{x}_k, \boldsymbol{u}_k) + \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}(\boldsymbol{x}_k, \boldsymbol{u}_k)\hat{\boldsymbol{\mu}}_t(\boldsymbol{\alpha}(\boldsymbol{x}_k)) \\ &+ \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k)\frac{\partial \hat{\boldsymbol{\mu}}_t(\boldsymbol{\alpha}(\boldsymbol{x}_k))}{\partial \boldsymbol{x}} \\ = &\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\boldsymbol{x}_k, \boldsymbol{u}_k) + \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}(\boldsymbol{x}_k, \boldsymbol{u}_k)\hat{\boldsymbol{A}}_t \boldsymbol{\varphi}(\boldsymbol{\alpha}(\boldsymbol{x}_k)) \\ &+ \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k)\hat{\boldsymbol{A}}_t \frac{\partial \boldsymbol{\varphi}}{\partial \boldsymbol{\alpha}}(\boldsymbol{\alpha}(\boldsymbol{x}_k))\frac{\partial \boldsymbol{\alpha}}{\partial \boldsymbol{x}}(\boldsymbol{x}_k). \end{aligned} \quad (24)$$

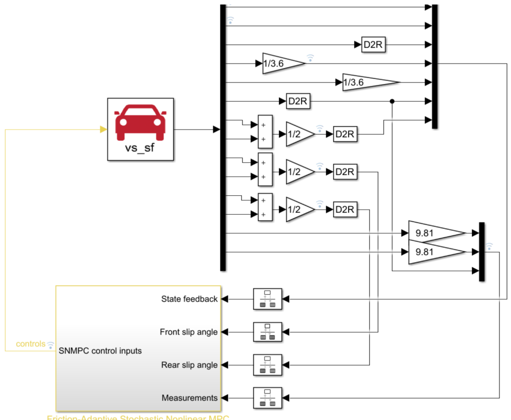

Fig. 1. CarSim simulation setup. Inputs to CarSim are the throttle position, the brake pedal position, as well as the steering angle command for the left and right front wheels. Outputs from CarSim are the longitudinal and lateral position, the yaw angle, the vehicle velocity, as well as the steering angle for left and right wheels. The controller block uses the vehicle state and the reference trajectory in order to compute a steering rate command as well as a torque command.

We momentarily define $\boldsymbol{g}_k := \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k)$, $\boldsymbol{\varphi}_k := \boldsymbol{\varphi}(\boldsymbol{\alpha}(\boldsymbol{x}_k))$ and approximate the covariance propagation as

$$\begin{aligned}\mathbb{E}[\boldsymbol{x}_{k+1}\boldsymbol{x}_{k+1}^\top] &\approx \mathbb{E}[(\boldsymbol{F}_k\boldsymbol{x}_k + \boldsymbol{g}_k\hat{\boldsymbol{A}}_t\boldsymbol{\varphi}_k)(\boldsymbol{F}_k\boldsymbol{x}_k + \boldsymbol{g}_k\hat{\boldsymbol{A}}_t\boldsymbol{\varphi}_k)^\top] \\ &= \boldsymbol{F}_k\boldsymbol{P}_k\boldsymbol{F}_k^\top + \mathbb{E}(\boldsymbol{g}_k\hat{\boldsymbol{A}}_t\boldsymbol{\varphi}_k\boldsymbol{\varphi}_k^\top\hat{\boldsymbol{A}}_t^\top\boldsymbol{g}_k^\top) \\ &= \boldsymbol{F}_k\boldsymbol{P}_k\boldsymbol{F}_k^\top + \underbrace{\boldsymbol{g}_k}_{\boldsymbol{G}_k}\underbrace{\operatorname{cov}(\hat{\boldsymbol{A}}_t\boldsymbol{\varphi}_k)}_{\boldsymbol{\Sigma}_k}\boldsymbol{g}_k^\top. \end{aligned} \tag{25}$$

Eq. (25) leads to the SNMPC formulation, where the integration of the tire-friction estimates is done by using the estimated mean in the state prediction (6c) and the estimated covariance function through the Jacobians in the covariance propagation (6d). Algorithm 1 summarizes our proposed friction-adaptive stochastic predictive control strategy.

Algorithm 1 Proposed SNMPC with Friction Adaptation

for each time step k **do**
- Estimate current state vector $\hat{\boldsymbol{x}}_k$, tire-friction estimate $\hat{\boldsymbol{A}}_k\boldsymbol{\varphi}(\boldsymbol{\alpha}(\boldsymbol{x}_k^e, \boldsymbol{u}_k^e))$ and covariance function $\boldsymbol{\Sigma}_k(\boldsymbol{\alpha}(\boldsymbol{x}_k^e, \boldsymbol{u}_k^e))$, using (22) (see [22]).
- Perform SQP iteration(s) to approximately solve SNMPC problem (6), using current state estimate in (6b), mean friction function in (6c), Jacobian (24) and covariance propagation (25) in (6d).

VI. CarSim Simulation Results

We validate the proposed friction-adaptive SNMPC in simulation using CarSim [10]. CarSim is a high-fidelity vehicle dynamics simulator, which we utilize to study the method's performance and applicability to a complex dynamical system—with a simplified model for controller design. Fig. 1 shows a block diagram of a vehicle in CarSim being controlled in a Simulink interface. Fig. 2 shows a screenshot of the CarSim environment, in which a vehicle performs lane changes while adapting its control parameters. For reproducibility, we use modules that are standard in CarSim for all components such as suspension, springs, tires, and so on, and the vehicle parameters are from a mid-size SUV.

Fig. 2. Illustration of a vehicle performing lane-change maneuvers in CarSim, while friction parameters and uncertainty are adapted online for SNMPC. The purple arrows show the lateral forces acting onto the wheels.

We consider a sequence of double lane-change maneuvers similar to the standardized ISO 3888-2 [24] double lane-change maneuver, starting on asphalt, switching to snow, and then back to asphalt. The reference is generated with Bezier polynomials and the position, heading, longitudinal velocity, and yaw rate are given to the controllers to track. For simplicity, we set the longitudinal velocity reference to be constant, $v_{\text{ref}}^X = 18$m/s. The different tuning parameters in the estimator are fairly generic, and the same as in [22]. We initialize the tire-friction estimates to correspond to the snow tire parameters in [25]. We evaluate the following controllers:

- PROPOSED: adaptive SNMPC method (Algorithm 1).
- SNOW: nominal NMPC that is based on fixed snow tire parameters using a Pacejka tire model [23].
- ASPHALT: nominal NMPC that is based on fixed asphalt tire parameters using a Pacejka tire model [23].

For the nominal NMPC using snow and asphalt parameters, we use the tire parameters from [25], which correspond to tire parameters originating from experimental validations. The metric to evaluate the controllers is the closed-loop cost Cost $= \sum_k l_k(\boldsymbol{x}_k, \boldsymbol{u}_k)$, over each time step, where $l_k(\cdot)$ is defined by (7).

A. Simulation Results

The performance of the estimator has been validated in several papers [8], [22]. Here, we focus on the closed-loop control performance. Fig. 3 shows a comparison of the lateral tracking performance between the three different controllers. The controller denoted by ASPHALT shows significant overshoot and poor tracking performance once the surface switches to snow (indicated by the gray area). The results demonstrate that the reference trajectory is tracked well by the PROPOSED and SNOW controllers, and that there are some transients in PROPOSED due to the convergence time of the tire-friction estimator. The SNOW controller performs conservatively as expected, and also has difficulty in quickly reacting to the reference change.

Fig. 4 displays the wheel steering angle and closed-loop cost for the three controllers. Note that the MPC cost function consists of more terms than the lateral tracking error, and focusing solely on the lateral position can lead to wrong conclusions about performance. Fig. 4 shows that the surface change at about 21s destabilizes the vehicle when using ASPHALT NMPC. Also, we can see that the PROPOSED

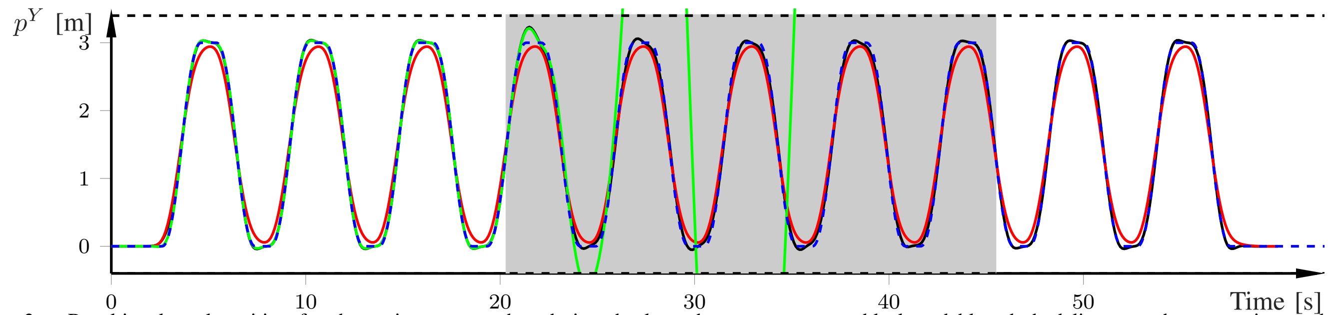

Fig. 3. Resulting lateral position for the various approaches during the lane-change maneuvers. black and blue dashed lines are the constraints and reference, respectively. The gray area indicates snow surface. The black, red, and green lines indicate the trajectories for PROPOSED, SNOW, and ASPHALT.

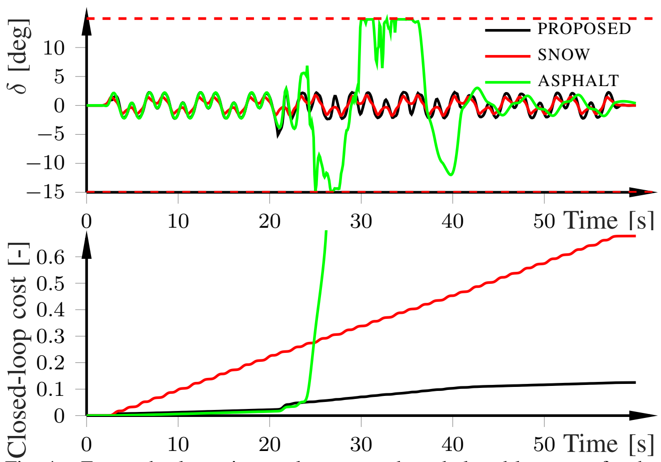

Fig. 4. Front wheel steering angle commands and closed-loop cost for the simulation results of the vehicle maneuver in Fig. 3.

adaptive SNMPC improves performance compared to the SNOW NMPC throughout the maneuver, reducing the closed-loop cost from over 0.6 to about 0.1.

VII. Conclusions

The presented approach integrates a recently developed Bayesian tire-friction estimator with an SNMPC formulation that includes uncertainty propagation, where the uncertainty is dependent on the estimated tire-friction distribution. The Bayesian formulation of the tire-friction estimation problem makes it possible to reason about uncertainty in the learning process. Hence, lack of knowledge about the road surface in certain regions is reflected in an increased uncertainty of the estimates in such regions. The results demonstrate that including uncertainty prediction indeed improves robustness compared to a nominal integration.

References

[1] K. Berntorp, "Path planning and integrated collision avoidance for autonomous vehicles," in *Amer. Control Conf.*, Seattle, WA, May 2017.

[2] S. Di Cairano and I. V. Kolmanovsky, "Real-time optimization and model predictive control for aerospace and automotive applications," in *Amer. Control Conf.*, Milwaukee, WI, Jun. 2018.

[3] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "Predictive active steering control for autonomous vehicle systems," *IEEE Trans. Control Syst. Technol.*, vol. 15, no. 3, pp. 566–580, 2007.

[4] K. Berntorp, R. Quirynen, T. Uno, and S. Di Cairano, "Trajectory tracking for autonomous vehicles on varying road surfaces by friction-adaptive nonlinear model predictive control," *Veh. Syst. Dyn.*, vol. 58, no. 5, pp. 705–725, 2019.

[5] K. Berntorp, B. Olofsson, K. Lundahl, and L. Nielsen, "Models and methodology for optimal trajectory generation in safety-critical road–vehicle manoeuvres," *Veh. Syst. Dyn.*, vol. 52, no. 10, pp. 1304–1332, 2014.

[6] R. Quirynen, K. Berntorp, and S. Di Cairano, "Embedded optimization algorithms for steering in autonomous vehicles based on nonlinear model predictive control," in *Amer. Control Conf.*, Milwaukee, WI, Jun. 2018.

[7] J. Svendenius, "Tire modeling and friction estimation," Ph.D. dissertation, Dept. Automatic Control, Lund University, Sweden, Apr. 2007.

[8] K. Berntorp, "Bayesian inference and learning of Gaussian-process state-space models," *Automatica*, vol. 129, p. 109613, 2021.

[9] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," in *Handbook of Nonlinear Filtering*, D. Crisan and B. Rozovsky, Eds. Oxford University Press, 2009.

[10] Mechanical Simulation, "CarSim," *https://www.carsim.com*, ver. 2021.0.

[11] Z. Ercan, M. Gokasan, and F. Borrelli, "An adaptive and predictive controller design for lateral control of an autonomous vehicle," in *Int. Conf. Vehicular Electronics and Safety*, Vienna, Austria, Feb. 2017.

[12] J. Wurts, J. Dallas, J. L. Stein, and T. Ersal, "Adaptive nonlinear model predictive control for collision imminent steering with uncertain coefficient of friction," in *Amer. Control Conf.*, Denver, CO, Jul. 2020.

[13] D. Telen, M. Vallerio, L. Cabianca, B. Houska, J. V. Impe, and F. Logist, "Approximate robust optimization of nonlinear systems under parametric uncertainty and process noise," *J. Proc. Control*, vol. 33, pp. 140–154, 2015.

[14] R. Quirynen and S. Di Cairano, "PRESAS: Block-structured preconditioning of iterative solvers within a primal active-set method for fast model predictive control," *Optimal Control Appl. Methods*, 2020.

[15] X. Feng, S. Di Cairano, and R. Quirynen, "Inexact adjoint-based SQP algorithm for real-time stochastic nonlinear MPC," in *IFAC World Congress*, Berlin, Germany, Jul. 2020.

[16] S. Vaskov, K. Berntorp, and R. Quirynen, "Cornering stiffness adaptive, stochastic nonlinear model predictive control for vehicles," in *Amer. Control Conf.*, New Orleans, LA, May 2021.

[17] K. Berntorp and S. Di Cairano, "Tire-stiffness and vehicle-state estimation based on noise-adaptive particle filtering," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 3, pp. 1100–1114, 2018.

[18] R. Rajamani, *Vehicle Dynamics and Control*. Springer-Verlag, 2006.

[19] M. Diehl, R. Findeisen, F. Allgöwer, H. G. Bock, and J. P. Schlöder, "Nominal stability of the real-time iteration scheme for nonlinear model predictive control," *IEE Proc.-Control Theory Appl.*, vol. 152, no. 3, pp. 296–308, 2005.

[20] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl., "Casadi–a software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, 2018.

[21] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed., ser. Operations Research and Financial Engineering. Springer, 2006.

[22] K. Berntorp, "Online Bayesian tire-friction learning by Gaussian-process state-space models," in *IFAC World Congress*, Berlin, Germany, Jul. 2020.

[23] H. B. Pacejka, *Tire and Vehicle Dynamics*, 2nd ed. Oxford, United Kingdom: Butterworth-Heinemann, 2006.

[24] ISO 3888-2: 2002, "Passenger cars–test track for a severe lane change manoeuvre–part 2: Obstacle avoidance," 2002.

[25] K. Berntorp, "Particle filtering and optimal control for vehicles and robots," Ph.D. dissertation, Dept. Automatic Control, Lund University, Sweden, May 2014.