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

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

Berntorp, K.

TR2019-051 June 29, 2019

Abstract

This paper addresses learning of the tire-friction curve for road vehicles, using a batch of wheel-speed and inertial measurements. We formulate a Bayesian approach based on recent advances in particle filtering and Markov chain Monte-Carlo methods. The unknown function mapping the wheel slip to tire friction is modeled as a Gaussian process (GP) that is included in a dynamic vehicle model relating the GP to the vehicle state. The approach is nonparametric and learns the probability density function of the tire friction, from which explicit estimates can be extracted. One benefit of the method is that it is not subject to overfitting issues. We illustrate the efficacy of the method for a set of simulated step-steer maneuvers. The results show that the method can accurately identify the nonlinear tire-friction curves, even for a limited amount of data.

European Control Conference (ECC)

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

Karl Berntorp

***Abstract*— This paper addresses learning of the tire-friction curve for road vehicles, using a batch of wheel-speed and inertial measurements. We formulate a Bayesian approach based on recent advances in particle filtering and Markov chain Monte-Carlo methods. The unknown function mapping the wheel slip to tire friction is modeled as a Gaussian process (GP) that is included in a dynamic vehicle model relating the GP to the vehicle state. The approach is nonparametric and learns the probability density function of the tire friction, from which explicit estimates can be extracted. One benefit of the method is that it is not subject to overfitting issues. We illustrate the efficacy of the method for a set of simulated step-steer maneuvers. The results show that the method can accurately identify the nonlinear tire-friction curves, even for a limited amount of data.**

I. Introduction

Advanced driver-assistance systems (ADAS) mainly actuate the vehicle through the tire–road contact. Knowledge of the tire–road relation is therefore of high importance in ADAS. The interaction between tire and road is highly nonlinear, and the parameters describing the nonlinear relation vary heavily between different surfaces and depend on several factors [1]. It is common to model the tire friction as a static function of the *slip*, and several different tire models are reported in [1]–[3]. Fig. 1 shows the typical tire-friction curves generated by the Pacejka (Magic formula) tire model [2]. The parametrizations used vary across the different models reported in literature, but the main characteristics are similar. Unfortunately, the vehicle states involved in the tire-friction estimation are not directly measured in production vehicles. Methods for identifying the parameters of the Brush model based on nonlinear optimization can be found in [1], [4]. The method in [5] uses an unscented Kalman filter (UKF) that augments the vehicle state and models the Pacejka parameters as random walk processes. The work [6] employs recursive least-squares for estimating the cornering stiffness (the linear slope of the friction curve), and [7] performs estimation of the Pacejka tire parameters by generating artificial data associated with different tire parameters and solving for the best fit to measured data. In [8], the Brush tire model and a nonlinear observer is used to estimate the peak friction coefficient under different excitation levels.

In this paper, we develop a fully Bayesian approach for identifying the tire-friction function assuming sensors available in production cars. We model the unknown function describing the tire friction as a Gaussian process [9], which combined with particle filtering (PF) [10] and Markov chain Monte-Carlo (MCMC) methods [11] results in a method for estimating the posterior density function (PDF) of the tire friction, given the measurement data. From the PDF, function estimates can be extracted, for instance, as the mean of the PDF. Since the method is nonparametric, it is not subject to specific modeling constraints that various tire models impose. Still, the method is insensitive to overfitting to the data.

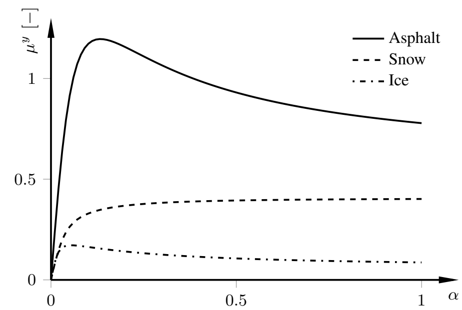

Fig. 1. Examples of lateral tire friction μ^y as a function of slip angle α for surfaces corresponding to asphalt, loose snow, and ice.

Gaussian processes (GPs) [9] are effective tools for nonparametric modeling of static nonlinear functions and has recently been extended to dynamical system behavior [12]. We leverage the recently developed framework in [13], where GP state-space models (GP-SSM) are used in a particle MCMC setting to learn general nonlinear SSMs. A straightforward application of the approach in [13] to our friction-curve identification is impractical, since the partial knowledge of the vehicle dynamics is not fully utilized. We therefore decompose the vehicle dynamics into an unknown part and a known part, where the unknown tire friction characteristics is modeled as a GP. By leveraging this decomposition, we formulate a method that can efficiently learn the input-output mapping between slip and tire friction.

Notation: With $p(\boldsymbol{x}_{0:t}|\boldsymbol{y}_{0:t})$, we mean the posterior density function of the state trajectory $\boldsymbol{x}_{0:t}$ from time index 0 to time index t given the measurement sequence $\boldsymbol{y}_{0:t} := \{\boldsymbol{y}_0, \dots, \boldsymbol{y}_t\}$. We define $\boldsymbol{f}_t := \boldsymbol{f}(\boldsymbol{x}_t)$ for a function $\boldsymbol{f}$. For a vector $\boldsymbol{x}$, $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ indicates that $\boldsymbol{x}$ is Gaussian distributed with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ and x_n denotes the nth component of $\boldsymbol{x}$. Matrices are indicated in capital bold font as $\boldsymbol{X}$, and the element on row i and column j is denoted with X_{ij}. The notation $\boldsymbol{f} \sim \mathcal{GP}(\boldsymbol{0}, \boldsymbol{\kappa}_{\theta,f}(\boldsymbol{x}, \boldsymbol{x}'))$ means that the function $\boldsymbol{f}(\boldsymbol{x})$ is a realization from a GP prior with a given covariance function $\boldsymbol{\kappa}_{\theta,f}(\boldsymbol{x}, \boldsymbol{x}')$ subject to hyperparameters $\boldsymbol{\theta}$, and $\mathcal{IW}(\nu, \boldsymbol{\Lambda})$ is the inverse-Wishart distribution with degree of freedom ν and scale matrix $\boldsymbol{\Lambda}$. Similarly, $\mathcal{MN}(\boldsymbol{M}, \boldsymbol{Q}, \boldsymbol{V})$ is the Matrix-Normal distribution with mean $\boldsymbol{M}$, right covariance $\boldsymbol{Q}$, and left precision $\boldsymbol{V}$.

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

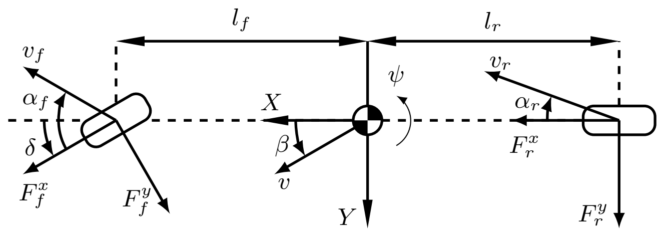

Fig. 2. A schematic of the single-track model and related notation.

II. Modeling and Problem Formulation

We use a single-track chassis model that includes the lateral velocity v^Y, yaw rate $\dot{\psi}$, and, optionally, the longitudinal velocity v^X as states, $\boldsymbol{x} = [v^X \ v^Y \ \dot{\psi}]^\mathrm{T} \in \mathbb{R}^d$. The single-track model, shown in Fig. 2, lumps together the left and right wheel on each axle, and roll and pitch dynamics are neglected. The model is standard. For the equations, see [14].

The tire friction components μ_i^x, μ_i^y, $i \in \{f, r\}$ are modeled as static functions of the slip quantities,

$$\mu_i^x = f_i^x(\lambda_i(\boldsymbol{x}), \alpha_i(\boldsymbol{x})), \tag{1a}$$

$$\mu_i^y = f_i^y(\alpha_i(\boldsymbol{x}), \lambda_i(\boldsymbol{x})), \tag{1b}$$

where λ is the slip ratio and α is the slip angle. These are defined as in [2],

$$\alpha_i = -\arctan\left(\frac{v_{y,i}}{v_{x,i}}\right), \tag{2}$$

$$\lambda_i = \frac{R_w \omega_i - v_{x,i}}{v_{x,i}}, \ i \in \{f, r\}, \tag{3}$$

where R_w is the known wheel radius, ω_i is the known wheel angular velocity for wheel i, and $v_{x,i}$ and $v_{y,i}$ are the longitudinal and lateral wheel velocities for wheel i with respect to an inertial system, expressed in the coordinate system of the wheel. The wheel velocities can be computed from a transformation of the longitudinal and lateral vehicle velocities. For brevity, we define the two vectors $\boldsymbol{\alpha} = [\alpha_f \ \alpha_r]^\mathrm{T}$, $\boldsymbol{\lambda} = [\lambda_f \ \lambda_r]^\mathrm{T}$ and the total slip vector as $\boldsymbol{s} = [\boldsymbol{\alpha}^\mathrm{T} \ \boldsymbol{\lambda}^\mathrm{T}]^\mathrm{T}$. The wheel dynamics are

$$T_i - I_w \dot{\omega}_i - F^z \mu_i^x R_w = 0 \ , \quad i \in \{f, r\}. \tag{4}$$

Here, T_i is the driving/braking torque for wheel i and I_w is the wheel inertia.

We compactly write (1) as

$$\boldsymbol{\mu} = \begin{bmatrix} f_f^x & f_r^x & f_f^y & f_r^y \end{bmatrix}^\mathrm{T}, \tag{5}$$

and model the friction vector as realizations from a Gaussian process prior

$$\boldsymbol{\mu}(\boldsymbol{s}) \sim \mathcal{GP}(\boldsymbol{0}, \boldsymbol{\kappa}_{\theta,\mu}(\boldsymbol{s}, \boldsymbol{s}')), \tag{6}$$

where the covariance function $\boldsymbol{\kappa}_{\theta,\mu}(\boldsymbol{s}, \boldsymbol{s}')$ is chosen in advance. In this work the hyperparameters $\boldsymbol{\theta}$ are determined a priori but can also be included in the learning process [15]. For completeness we include both longitudinal components in (5). However, in general this will lead to poor observability of the longitudinal components without further assumptions, such as front/rear wheel drive or additional sensing.

A. Estimation Model

After discretization with sampling period T_s and using $\boldsymbol{u} = [\delta \ \omega_i]^\mathrm{T}$ as the known input vector, the vehicle model can be written as

$$\boldsymbol{x}_{t+1} = \boldsymbol{a}(\boldsymbol{x}_t, \boldsymbol{u}_t) + \boldsymbol{G}(\boldsymbol{x}_t, \boldsymbol{u}_t)\boldsymbol{\mu}(\boldsymbol{s}_t), \tag{7}$$

where $\boldsymbol{a}$ and $\boldsymbol{G}$ are the (known) parts of the vehicle model, and $\boldsymbol{\mu}$ is the unknown function. Learning of $\boldsymbol{\mu}$ amounts to estimating the distribution $p(\boldsymbol{\mu}|\boldsymbol{y}_{0:T})$ given the measurement history $\boldsymbol{y}_{0:T}$.

Our sensor configuration is based on a setup commonly available in production cars, namely the longitudinal and lateral accelerations, a_m^X, a_m^Y, and the yaw rate $\dot{\psi}_m$, forming the measurement vector $\boldsymbol{y} = [a_m^X \ a_m^Y \ \dot{\psi}_m]^\mathrm{T}$. The yaw-rate measurement is directly related to the yaw rate, whereas a^X and a^Y can be extracted from the single-track model. We model the measurement noise $\boldsymbol{e}_t$ as zero-mean Gaussian distributed noise with covariance $\boldsymbol{R}_t$ according to $\boldsymbol{e}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}_t)$. The measurement model can be written as

$$\boldsymbol{y}_t = \boldsymbol{h}(\boldsymbol{x}_t, \boldsymbol{u}_t) + \boldsymbol{D}(\boldsymbol{x}_t, \boldsymbol{u}_t)\boldsymbol{\mu}(\boldsymbol{s}_t) + \boldsymbol{e}_t. \tag{8}$$

Similar to (7), the measurement model (8) is decomposed into known parts of the dynamics, $\boldsymbol{h}$ and $\boldsymbol{D}$, and an unknown part, $\boldsymbol{\mu}$. 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.

The estimation model consisting of (7) and (8) is a GP-SSM where the tire friction is a GP. The reason for modeling the tire friction as a GP is its ability to model the inherent uncertainty stemming from the measurement data, not only the uncertainty from the stochastic noise term $\boldsymbol{e}_t$, which affects the estimation quality, but also that the measurement data may contain few measurements in certain regions of the state space.

Remark 1: Automotive-grade inertial sensors usually have bias that affects the measurement quality. Since the proposed method uses batches of measurement data, we assume that the bias has already been removed (e.g., using [16]). If not, it can be directly incorporated into the framework (c.f. [17]).

B. Problem Formulation

We want to estimate the nonlinear function $\boldsymbol{\mu}$ describing the tire friction. We approach this problem as follows. Given the system model (7), (8), and a Gaussian process prior (6) on the tire friction, we want to infer the posterior distributions of $\boldsymbol{\mu}(\boldsymbol{s})$ given a set of measurement data $\boldsymbol{y}_{0:T}$,

$$p(\boldsymbol{\mu}|\boldsymbol{y}_{0:T}). \tag{9}$$

Since the tire friction estimate will depend on the vehicle state, we solve for (9) by approximating the joint posterior

$p(\boldsymbol{\mu}, \boldsymbol{x}_{0:T}|\boldsymbol{y}_{0:T})$ and perform the marginalization step

$$
\begin{aligned}
p(\boldsymbol{\mu}|\boldsymbol{y}_{0:T}) &= \int p(\boldsymbol{\mu}, \boldsymbol{x}_{0:T}|\boldsymbol{y}_{0:T}) \, \mathrm{d}\boldsymbol{x}_{0:T} \\
&= \int p(\boldsymbol{\mu}|\boldsymbol{x}_{0:T}, \boldsymbol{y}_{0:T}) p(\boldsymbol{x}_{0:T}|\boldsymbol{y}_{0:T}) \, \mathrm{d}\boldsymbol{x}_{0:T} \quad (10)
\end{aligned}
$$

to recover (9).

III. Reduced-Rank GP-SSMs and Particle Filtering

We rely on GP priors for learning the function describing the tire friction, where the covariance function $\boldsymbol{\kappa}(\boldsymbol{x}, \boldsymbol{x}')$ describes the prior assumptions. A bottleneck in some of the GP-SSM methods proposed in literature is the computational load. In this paper we use the computationally efficient reduced-rank GP-SSM framework presented in [13], [18]. For a thorough derivation and convergence proofs, see [18].

Following the notation in [18], isotropic covariance functions (i.e., they only depend on the Euclidean norm $\|\boldsymbol{x}-\boldsymbol{x}'\|$) can be approximated in terms of Laplace operators on the following form:

$$
\kappa_\theta(\boldsymbol{x}, \boldsymbol{x}') \approx \sum_{j_1,\ldots,j_d=1}^{m} \mathcal{S}_\theta(\lambda^{j_1,\ldots,j_d}) \phi^{j_1,\ldots,j_d}(\boldsymbol{x}) \phi^{j_1,\ldots,j_d}(\boldsymbol{x}'), \quad (11)
$$

where we for simplicity have assumed m basis functions for each state dimension. In (11), $\mathcal{S}_\theta$ is the spectral density of κ_θ and

$$
\lambda^{j_1,\ldots,j_d} = \sum_{n=1}^{d} \left(\frac{\pi j_n}{2L_n}\right)^2, \quad (12a)
$$

$$
\phi^{j_1,\ldots,j_d} = \prod_{n=1}^{d} \frac{1}{\sqrt{L_n}} \sin\left(\frac{\pi j_n (x_n + L_n)}{2L_n}\right), \quad (12b)
$$

are the Laplace operator eigenvalues and eigenfunctions, respectively. For brevity, we will in the rest of the paper denote $j_1, \ldots, j_d$ with $\boldsymbol{j}$.

From the approximation (11) using Laplace operators, [18] provides a relation between basis function expansions of a function f and GPs based on the Karhunen-Loeve expansion. With the basis functions chosen as (12b),

$$
f(\boldsymbol{x}) \sim \mathcal{GP}(0, \kappa(\boldsymbol{x}, \boldsymbol{x}')) \Leftrightarrow f(\boldsymbol{x}) \approx \sum_{\boldsymbol{j}} \gamma^{\boldsymbol{j}} \phi^{\boldsymbol{j}}(\boldsymbol{x}), \quad (13)
$$

with

$$
\gamma^{\boldsymbol{j}} \sim \mathcal{N}(0, S(\lambda^{\boldsymbol{j}}). \quad (14)
$$

For a state-space model $\boldsymbol{x}_{t+1} = \boldsymbol{f}(\boldsymbol{x}_t) + \boldsymbol{w}_t$, (13) implies the reduced-rank GP-SSM

$$
\boldsymbol{x}_{t+1} = \begin{bmatrix} \gamma_1^1 & \cdots & \gamma_1^m \\ \vdots & & \vdots \\ \gamma_d^1 & \cdots & \gamma_d^m \end{bmatrix} \begin{bmatrix} \phi^1(\boldsymbol{x}_t) \\ \vdots \\ \phi^m(\boldsymbol{x}_t) \end{bmatrix} + \boldsymbol{w}_t, \quad (15)
$$

where γ_n^j are the weights to be learned, m is the total number of basis functions (i.e., m^d in (11)), and $\boldsymbol{w}_t$ is zero-mean Gaussian distributed noise with covariance $\boldsymbol{Q}$. In Sec. IV, (15) in combination with PF forms the basis for learning the tire friction.

A. Sequential Monte Carlo and Particle Filtering

Sequential Monte-Carlo (SMC) methods, such as PFs, constitute a class of techniques that estimate the posterior distribution in SSMs, and SMCs have recently emerged as a useful tool in learning of SSMs (e.g., [19]). PFs approximate the posterior density $p(\boldsymbol{x}_t|\boldsymbol{y}_{0:t})$ by a set of N weighted state trajectories as

$$
p(\boldsymbol{x}_t|\boldsymbol{y}_{0:t}) \approx \sum_{i=1}^{N} q_t^i \delta_{\boldsymbol{x}_t^i}(\boldsymbol{x}_t), \quad (16)
$$

where q_t^i is the importance weight of the ith particle $\boldsymbol{x}_t^i$ and $\delta(\cdot)$ is the Dirac delta mass. The PF recursively estimates (16) by repeated application of Bayes' rule, where the states are sampled according to a proposal density $\pi(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{y}_t))$, which in the simplest case is the dynamical model. This yields the state trajectory samples at each time step as

$$
\boldsymbol{x}_t^i \sim p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}^i), \quad i \in \{1, \ldots, N\}. \quad (17)
$$

The importance weights are updated using the likelihood as

$$
q_t^i \propto q_{t-1}^i p(\boldsymbol{y}_t|\boldsymbol{x}_t^i). \quad (18)
$$

The PF algorithm iterates between (17) and (18), combined with a resampling step that removes particles with low weights and replaces them with more likely particles.

When using SMC for learning (e.g., parameters), the PF is used to repeatedly sample state trajectories $\boldsymbol{x}_{0:T}$ within an MCMC procedure, combined with updating the parameters to fit the trajectory $\boldsymbol{x}_{0:T}$ from the previous iteration followed by sampling the parameters to be used in the next iteration of the PF. The idea is that as the number of iterations k grow large, the sampled state trajectories and parameters are indeed samples from the correct distribution.

We adapt a conditional PF with ancestor sampling (CPF-AS) [11] to generate the state trajectories needed to learn the function $\boldsymbol{\mu}$ describing the tire friction. CPF-AS generates the state trajectories by a procedure similar to the standard PF, except for that the PF is conditioned on one prespecified *reference trajectory* $\boldsymbol{x}_{0:T}'$, which is retained throughout the procedure. When used within an MCMC procedure [20], it can be shown that after a burn-in period, the state trajectories generated by CPF-AS are samples drawn from the smoothing distribution $p(\boldsymbol{x}_{0:T}|\boldsymbol{y}_{0:T})$ for any finite $N > 1$ [11], [21], that is, the second distribution on the right-hand side of (10).

IV. Learning the Tire Friction by GP-SSMs

The objective is to infer the posterior distribution (9) of the unknown function $\boldsymbol{\mu}$. In the presentation of our method, for brevity we will focus on the lateral dynamics, that is, we learn the lateral tire friction of front and rear wheels. Note that the extension to the longitudinal dynamics is analogous.

A. Adapting the Model for Learning

The Bayesian learning method we leverage assumes dynamical systems on the form $\boldsymbol{x}_{t+1} = \boldsymbol{f}_t + \boldsymbol{w}_t$, where the full state-transition function $\boldsymbol{f}_t$ is to be learned. Hence, we need to adapt the vehicle model (7). Specifically, by manipulation of (7), it is possible to show that

$$\mu_r^y(\boldsymbol{s}_t) = \frac{G_{11,t}}{G_{12,t}G_{22,t} - G_{12,t}G_{23,t}}\bar{x}_{2,t+1} - \frac{G_{21,t}}{G_{11,t}G_{22,t} - G_{12,t}G_{21,t}}\bar{x}_{1,t+1}, \quad (19)$$

where $\bar{\boldsymbol{x}}_{t+1} = \boldsymbol{x}_{t+1} - \boldsymbol{a}_t$. Similarly,

$$\mu_f^y(\boldsymbol{s}_t) = \left(\frac{1}{G_{11,t}} + \frac{G_{12,t}G_{21,t}}{G_{11,t}G_{22,t} - G_{12,t}G_{23,t}}\right)\bar{x}_{1,t+1} - \frac{G_{12,t}}{G_{11,t}G_{22,t} - G_{12,t}G_{21,t}}\bar{x}_{2,t+1}. \quad (20)$$

We introduce the shorthand notation $\boldsymbol{\zeta}_t = [\zeta_{1,t}\ \zeta_{2,t}]^{\mathrm{T}}$ for the right-hand sides of (19), (20), which results in the system

$$\boldsymbol{\zeta}_{t+1} = \boldsymbol{\mu}(\boldsymbol{s}_t) + \boldsymbol{w}_t, \quad (21)$$

where $\boldsymbol{w}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q})$ is zero mean Gaussian distributed noise with, possibly unknown, covariance matrix $\boldsymbol{Q}$, which accounts for modeling errors.

Using the basis function expansion approach (13), we formulate a reduced-rank GP-SSM of (21) similar to (15), which results in

$$\boldsymbol{\zeta}_{t+1} = \underbrace{\begin{bmatrix} \gamma_1^1 & \cdots & \gamma_1^m \\ \vdots & & \vdots \\ \gamma_d^1 & \cdots & \gamma_d^m \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} \phi^1(\boldsymbol{s}_t) \\ \vdots \\ \phi^m(\boldsymbol{s}_t) \end{bmatrix}}_{\boldsymbol{\varphi}(\boldsymbol{s}_t)} + \boldsymbol{w}_t. \quad (22)$$

B. Tire Friction Learning with GP-SSM

With the reduced-rank GP-SSM (22) in combination with the measurement model (8), we are now ready to formulate our learning approach. With the GP-SSM, the problem of estimating the distribution (9) now amounts to infer the distribution of $\boldsymbol{A}$ and $\boldsymbol{Q}$, that is, to estimate the distribution

$$p(\boldsymbol{A}, \boldsymbol{Q} | \boldsymbol{y}_{0:T}), \quad (23)$$

where the components in $\boldsymbol{A}$ are Gaussian distributed according to (14). To estimate the covariance matrix $\boldsymbol{Q}$, we impose the additional assumption that the prior of $\boldsymbol{Q}$ is inverse-Wishart ($\mathcal{IW}$) distributed according to

$$\boldsymbol{Q} \sim \mathcal{IW}(\ell_Q, \boldsymbol{\Lambda}_Q). \quad (24)$$

The $\mathcal{IW}$ distribution is a distribution over (real) positive definite matrices, and has the degrees of freedom ℓ_Q and positive definite scale matrix $\boldsymbol{\Lambda}_Q$ as hyperparameters. Letting an unknown covariance matrix have the $\mathcal{IW}$ distribution as prior distribution is common due to its properties and has been done in automotive applications before (e.g., [14], [22]).

The components of the system matrix $\boldsymbol{A}$ in (23), defined in (22), are Gaussian distributed (see (14)). Hence, $\boldsymbol{A}$ is Matrix-Normal ($\mathcal{MN}$) distributed according to

$$\boldsymbol{A} \sim \mathcal{MN}(\boldsymbol{0}, \boldsymbol{Q}, \boldsymbol{V}). \quad (25)$$

With $\boldsymbol{A}$ $\mathcal{MN}$ distributed and $\boldsymbol{Q}$ $\mathcal{IW}$ distributed, the joint prior $p(\boldsymbol{A}, \boldsymbol{Q})$ is $\mathcal{MNIW}$ distributed according to [23]

$$p(\boldsymbol{A}, \boldsymbol{Q}) = \mathcal{MNIW}(\boldsymbol{A}, \boldsymbol{Q} | \boldsymbol{0}, \boldsymbol{V}, \ell_Q, \boldsymbol{\Lambda}_Q), \quad (26)$$

where $\boldsymbol{V}$ has the inverse spectral density of the covariance function as diagonal entries [13],

$$\boldsymbol{V} = \mathrm{diag}(\begin{bmatrix} S^{-1}(\lambda^1) & \cdots & S^{-1}(\lambda^m) \end{bmatrix}), \quad (27)$$

and where $\mathrm{diag}(\cdot)$ is the diagonal matrix.

To estimate (23), we need the two densities $p(\boldsymbol{A}, \boldsymbol{Q} | \boldsymbol{x}_{0:T}, \boldsymbol{y}_{0:T})$ and $p(\boldsymbol{x}_{0:T} | \boldsymbol{y}_{0:T})$ similar to the right-hand side in (10).

1) Estimating the State Posterior: The state posterior $p(\boldsymbol{x}_{0:T} | \boldsymbol{y}_{0:T})$ will depend on the tire friction estimate through (7), which implies

$$p(\boldsymbol{x}_{0:T} | \boldsymbol{y}_{0:T}) = \int p(\boldsymbol{x}_{0:T} | \boldsymbol{A}, \boldsymbol{Q}, \boldsymbol{y}_{0:T}) \,\mathrm{d}\boldsymbol{A}, \mathrm{d}\boldsymbol{Q}. \quad (28)$$

Hence, we need to sample from $p(\boldsymbol{x}_{0:T} | \boldsymbol{A}, \boldsymbol{Q}, \boldsymbol{y}_{0:T})$. We use CPF-AS, outlined in Algorithm 1, which produces samples that are asymptotically consistent with (28) when encapsulated into an MCMC procedure [11].

Algorithm 1 CPF-AS

Input: $\boldsymbol{x}_{0:T}(k)$, $\boldsymbol{u}_{0:T-1}$, N, model $\{\boldsymbol{a}, \boldsymbol{G}, \boldsymbol{\mu}, \boldsymbol{Q}, \boldsymbol{D}, \boldsymbol{R}\}$.
Output Trajectory $\boldsymbol{x}_{0:T}(k+1)$.

```
Sample x_1^i ~ p(x_1),   ∀i ∈ {1, ..., N−1}.
Set x_1^N = x_1(l).
for t ← 1 to T do
    Compute s_t^i from (2) ∀i ∈ {1, ..., N}.
    Set q_t^i ∝ N(y_t | h_t^i + D_t^i μ(s_t^i), R),   ∀i ∈ {1, ..., N}.
    Sample a_t^i with P(a_t^i = j) ∝ q_t^j,   ∀i ∈ {1, ..., N}.
    Sample x_{t+1}^i ~ N(a_t^{a_t^i} + G_t^{a_t^i} μ(s_t^{a_t^i}), Q),
    ∀i ∈ {1, ..., N}.
    Set x_{t+1}^N = x_{t+1}(k).
    Sample a_t^N P(a_t^N = j) ∝ q_t^j N(x_{t+1}^N | a_t^j + G_t^j μ(s_t^j), Q).
    Set x_{1:t+1}^i = {x_{1:t}^{a_t^i}, x_{t+1}^i},   ∀i ∈ {1, ..., N}.
end for
Draw J with P(i = J) ∝ q_T^i.
Set x_{0:T}(k+1) = x_{0:T}^J.
```

2) Learning the Tire Friction: To learn the posterior (23), that is, to learn the PDF of the function describing the tire friction and process-noise covariance that accounts for modeling errors, we use Bayes' rule,

$$p(\boldsymbol{A}, \boldsymbol{Q} | \boldsymbol{x}_{0:T}, \boldsymbol{y}_{0:T}) \propto p(\boldsymbol{x}_{0:T}, \boldsymbol{y}_{0:T} | \boldsymbol{A}, \boldsymbol{Q}) p(\boldsymbol{A}, \boldsymbol{Q}). \quad (29)$$

The likelihood $p(\boldsymbol{x}_{0:T}, \boldsymbol{y}_{0:T} | \boldsymbol{A}, \boldsymbol{Q})$ can be written as

$$p(\boldsymbol{x}_{0:T}, \boldsymbol{y}_{0:T} | \boldsymbol{A}, \boldsymbol{Q}) = p(\boldsymbol{x}_0) \underbrace{\prod_{t=0}^{T-1} p(\boldsymbol{x}_{t+1} | \boldsymbol{x}_t, \boldsymbol{A}, \boldsymbol{Q})}_{p(\boldsymbol{x}_{0:T}|\boldsymbol{A}, \boldsymbol{Q})} \underbrace{\prod_{t=0}^{T} p(\boldsymbol{y}_t | \boldsymbol{x}_t, \boldsymbol{A}, \boldsymbol{Q})}_{p(\boldsymbol{y}_{0:T}|\boldsymbol{x}_{0:T}, \boldsymbol{A}, \boldsymbol{Q})}. \quad (30)$$

Conditioned on $\boldsymbol{A}$ and $\boldsymbol{Q}$, the vehicle model (7) and measurement model (8) are both Gaussian distributed, which implies that the two terms $p(\boldsymbol{x}_{0:T} | \boldsymbol{A}, \boldsymbol{Q})$ and $p(\boldsymbol{y}_{0:T} | \boldsymbol{x}_{0:T}, \boldsymbol{A}, \boldsymbol{Q})$ in (30) are Gaussian distributed. Hence, the density $p(\boldsymbol{x}_{0:T}, \boldsymbol{y}_{0:T}|\boldsymbol{A}, \boldsymbol{Q})$ is Gaussian distributed since it is a product of Gaussian distributions. Therefore, we can utilize the concept of conjugate priors. If a prior distribution belongs to the same family as the posterior distribution, the prior is conjugate to the likelihood. For Gaussian distributed data, an $\mathcal{MNIW}$ distribution defines a conjugate prior [24]. This is convenient since it allows closed-form expressions for the update of $\boldsymbol{A}$ and $\boldsymbol{Q}$ [13]. Let us define

$$\boldsymbol{\Phi} = \sum_{t=0}^{T} \boldsymbol{\zeta}_t \boldsymbol{\zeta}_t^{\mathrm{T}}, \tag{31a}$$

$$\boldsymbol{\Psi} = \sum_{t=0}^{T} \boldsymbol{\zeta}_t \boldsymbol{\varphi}(\boldsymbol{s}_t)^{\mathrm{T}}, \tag{31b}$$

$$\boldsymbol{\Sigma} = \sum_{t=0}^{T} \boldsymbol{\varphi}(\boldsymbol{s}_t)\boldsymbol{\varphi}(\boldsymbol{s}_t)^{\mathrm{T}}. \tag{31c}$$

Then it follows that the joint posterior is

$$p(\boldsymbol{A}, \boldsymbol{Q}|\boldsymbol{x}_{0:T}, \boldsymbol{y}_{0:T}) = p(\boldsymbol{A}|\boldsymbol{Q}, \boldsymbol{x}_{0:T}, \boldsymbol{y}_{0:T})p(\boldsymbol{Q}|\boldsymbol{x}_{0:T}, \boldsymbol{y}_{0:T}), \tag{32}$$

where

$$p(\boldsymbol{Q}|\boldsymbol{x}_{0:T}, \boldsymbol{y}_{0:T}) = \mathcal{IW}(\boldsymbol{Q}|T + \ell_Q, \boldsymbol{\Lambda}_Q + \boldsymbol{\Phi} - \boldsymbol{\Psi}(\boldsymbol{\Sigma} + \boldsymbol{V})^{-1}\boldsymbol{\Psi}^{\mathrm{T}}), \tag{33}$$

$$p(\boldsymbol{A}|\boldsymbol{Q}, \boldsymbol{x}_{0:T}, \boldsymbol{y}_{0:T}) = \mathcal{MN}(\boldsymbol{A}|\boldsymbol{\Psi}(\boldsymbol{\Sigma} + \boldsymbol{V})^{-1}, \boldsymbol{Q}, (\boldsymbol{\Sigma} + \boldsymbol{V})^{-1}). \tag{34}$$

Using Algorithm 1 combined with (33) and (34) for sampling the process-noise covariance $\boldsymbol{Q}$ and weight matrix $\boldsymbol{A}$ in the basis function expansion (13), the complete learning algorithm can be summarized as in Algorithm 2. Algorithm 2 generates the weight matrix $\boldsymbol{A}$ at each iteration k, which converges after a transient period. Denote the number of samples in the transient phase (burn-in period) with K_{bi}. Then $K - K_{\mathrm{bi}} - 1$ samples generated by Algorithm 2 at Line 9 are samples from the distribution $p(\boldsymbol{A}|\boldsymbol{Q}, \boldsymbol{x}_{0:T}, \boldsymbol{y}_{0:T})$, which implies that $\boldsymbol{\mu} = A(k+1)\boldsymbol{\varphi}$ is a sample from (9). To recover (9), we can write

$$p(\boldsymbol{\mu}|\boldsymbol{y}_{0:T}) \approx \frac{1}{K - K_{\mathrm{bi}} - 1} \sum_{k=K_{\mathrm{bi}}}^{K-1} \delta(\boldsymbol{\mu} - \boldsymbol{A}(k+1)\boldsymbol{\varphi}(\boldsymbol{s}_{0:T})), \tag{35}$$

and a function estimate can be determined as the posterior mean,

$$\hat{\boldsymbol{\mu}}(\boldsymbol{s}_t) = \underbrace{\frac{1}{K - K_{\mathrm{bi}} - 1} \sum_{k=K_{\mathrm{bi}}}^{K-1} \boldsymbol{A}(k+1)}_{\hat{\boldsymbol{A}}} \boldsymbol{\varphi}(\boldsymbol{s}_t), \tag{36}$$

V. Simulation Results

We evaluate Algorithm 2 on synthetic data, by simulating step-steer maneuvers at constant forward velocity using a nonlinear single-track model with the Pacejka tire model [2],

$$\begin{aligned} F_f^y &= \mu_f^y \sin(C_f^y \arctan(B_f^y(1 - E_f^y)\alpha_f + E_f^y \arctan(B_f^y \alpha_f))), \\ F_r^y &= \mu_r^y \sin(C_r^y \arctan(B_r^y(1 - E_r^y)\alpha_r + E_r^y \arctan(B_r^y \alpha_r))). \end{aligned} \tag{37}$$

The aggressiveness of the step-steer maneuver is such that the slip angles reach the saturated region. The vehicle chassis model uses parameters corresponding to a mid-size SUV and the tire parameters are taken from [25].

We use 10 basis functions each for the front and rear tire, which gives $m = 100$ basis functions in total. The simulation lasts for 500 time steps, corresponding to a data set lasting for 100 s with a sampling time of $T_s = 200$ ms. The number of samples in Algorithm 2 is set to $K = 10000$ and the burn-in period is set to $K_{\mathrm{bi}} = K/2$. The number of particles in the underlying PF is $N = 50$.

Fig. 3 shows the estimated lateral tire-friction curve of the front tire and Fig. 4 shows the equivalent of the rear tire. The excitation range of the underlying slip angles of the data is indicated by the bars in the lower plot. We stress that the underlying data are *not* used for learning. Where there is enough excitation, the proposed method can correctly learn the shape of the tire curve, while the estimate in the unobserved parts of the state space has large uncertainty, as expected. However, due to the zero-mean prior of the function coefficients in (14), the estimates do not suffer from overfitting issues outside of the available data range. It is highly likely that using more data improves the estimates further and will decrease the uncertainty.

Algorithm 2 Proposed method for tire-friction learning

Input: $\boldsymbol{y}_{0:T}$, $\boldsymbol{u}_{0:T-1}$, priors (26), (27).
Output K MCMC samples from $p(\boldsymbol{A}, \boldsymbol{Q}, \boldsymbol{x}_{0:T}|\boldsymbol{y}_{0:T})$
1: Sample initial guess $\boldsymbol{x}_{0:T}(0)$, $\boldsymbol{Q}(0)$, $\boldsymbol{A}(0)$.
2: **for** $k \leftarrow 0$ to $K-1$ **do**
3: Sample $\boldsymbol{x}_{0:T}(k+1)$ given $\boldsymbol{Q}(k)$, $\boldsymbol{A}(k)$ using Algorithm 1.
4: Compute $\boldsymbol{s}_{0:T}$ using (2).
5: Compute $\boldsymbol{\varphi}(\boldsymbol{s}_{0:T})$ in (22) using (12b).
6: Compute $\boldsymbol{\zeta}_{0:T}$ using (19)–(21).
7: Compute $\boldsymbol{\Phi}$, $\boldsymbol{\Psi}$, $\boldsymbol{\Sigma}$ using (31).
8: Sample $\boldsymbol{Q}(k+1)$ given $\boldsymbol{A}(k)$, $\boldsymbol{x}_{0:T}(k+1)$ using (33).
9: Sample $\boldsymbol{A}(k+1)$ given $\boldsymbol{x}_{0:T}(k+1)$, $\boldsymbol{Q}(k+1)$ using (34).
10: Set $\boldsymbol{\mu} = \boldsymbol{A}(k+1)\boldsymbol{\varphi}$.
11: **end for**

VI. Conclusion

We have presented a novel method for learning the nonlinear function describing the dependence between wheel slip and tire friction. The method is fully Bayesian and is based on recent developments in particle MCMC and GP-SSMs. A key feature is that the method only uses inertial and wheel-speed sensors, which are typically installed in production vehicles. This makes the method useful for learning in existing vehicle setups. The evaluation was done using synthetic data, using sensor noise values typical for automotive-grade sensors. The results indicate that the proposed method can indeed learn the nonlinear tire-friction curve in a Bayesian framework. The nonparametric nature of the method makes it flexible.

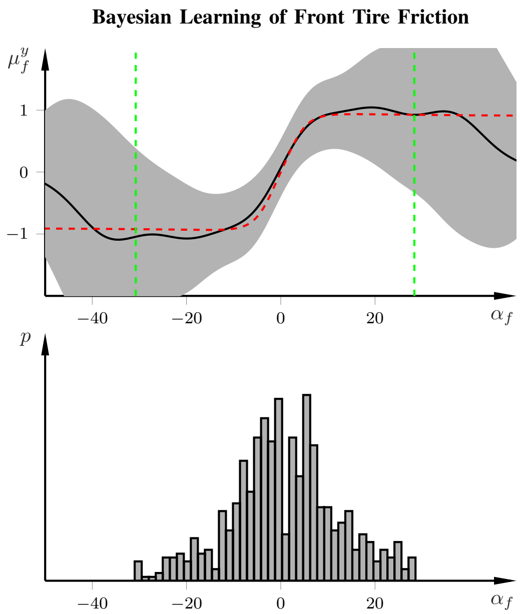

Fig. 3. Bayesian learning of the tire friction of the front tire (upper plot) with inertial sensing, and the distribution of the underlying data (lower plot). The black line is the posterior mean (36) of $p(\boldsymbol{\mu}|\boldsymbol{y}_{0:T})$ generated from Algorithm 2, and the 2σ lines are indicated by the shadowed areas. The red dashed curve is the true friction curve, and the green vertical dashed lines indicate the range of the underlying data (c.f. lower plot). The bars in the lower plot show the distribution of the slip angle of the front wheel computed from the underlying data points. Note that this data is not available for learning.

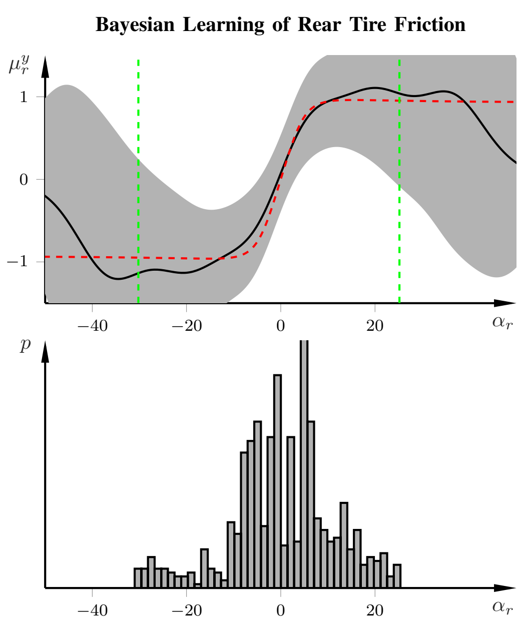

Fig. 4. Bayesian learning of the tire friction of the rear tire (upper plot) with inertial sensing, and the distribution of the underlying data (lower plot). Same notation as in Fig. 3.

References

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

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

[3] U. Kiencke and L. Nielsen, *Automotive Control Systems—For Engine, Driveline and Vehicle*, 2nd ed. Berlin Heidelberg: Springer-Verlag, 2005.

[4] B. Schofield, "Model-based vehicle dynamics control for active safety," phdthesis, Dept. Autom. Control, Lund University, 2008.

[5] B. Goldfain, P. Drews, C. You, M. Barulic, O. Velev, P. Tsiotras, and J. M. Rehg, "Autorally: An open platform for aggressive autonomous driving," *arXiv preprint arXiv:1806.00678*, 2018.

[6] C. Lundquist and T. B. Schön, "Recursive identification of cornering stiffness parameters for an enhanced single track model," in *15th IFAC Symp. System Identification*, Saint-Malo, France, July 2009.

[7] S. Garatti and S. Bittanti, "Parameter estimation in the Pacejka's tyre model through the TS method," in *IFAC Symp. System Identification*, Saint-Malo, France, July 2009.

[8] C. Ahn, H. Peng, and H. E. Tseng, "Robust estimation of road friction coefficient using lateral and longitudinal vehicle dynamics," *Veh. Syst. Dyn.*, vol. 50, no. 6, pp. 961–985, 2012.

[9] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.

[10] 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.

[11] F. Lindsten, M. I. Jordan, and T. B. Schön, "Particle Gibbs with ancestor sampling." *J. Machine Learning Res.*, vol. 15, no. 1, pp. 2145–2184, 2014.

[12] C. L. C. Mattos, Z. Dai, A. Damianou, J. Forth, G. A. Barreto, and N. D. Lawrence, "Recurrent Gaussian processes," *arXiv preprint arXiv:1511.06644*, 2015.

[13] A. Svensson, A. Solin, S. Särkkä, and T. B. Schön, "Computationally efficient Bayesian learning of Gaussian process state space models," in *Int. Conf. Artificial Intelligence and Statistics*, Cadiz, Spain, May 2016.

[14] K. Berntorp and S. Di Cairano, "Tire-stiffness and vehicle-state estimation based on noise-adaptive particle filtering," *IEEE Trans. Control Syst. Technol.*, 2018, in press.

[15] A. Svensson and T. B. Schön, "A flexible state-space model for learning nonlinear dynamical systems," *Automatica*, vol. 80, pp. 189–199, 2017.

[16] K. Berntorp and S. Di Cairano, "Offset and noise estimation of automotive-grade sensors using adaptive particle filtering," in *Amer. Control Conf.*, Milwaukee, WI, June 2018.

[17] ——, "Particle Gibbs with ancestor sampling for identification of tire-friction parameters," in *IFAC World Congress*, Toulouse, France, July 2017.

[18] A. Solin and S. Särkkä, "Hilbert space methods for reduced-rank Gaussian process regression," *arXiv preprint arXiv:1401.5508*, 2014.

[19] N. Kantas, A. Doucet, S. S. Singh, J. Maciejowski, N. Chopin, *et al.*, "On particle methods for parameter estimation in state-space models," *Statistical science*, vol. 30, no. 3, pp. 328–351, 2015.

[20] C. Robert and G. Casella, *Monte Carlo statistical methods*. Springer, 2013.

[21] C. Andrieu, A. Doucet, and R. Holenstein, "Particle Markov chain Monte Carlo methods," *J. Royal Statistical Society: Series B (Statistical Methodology)*, vol. 72, no. 3, pp. 269–342, 2010.

[22] E. Özkan, V. Šmídl, S. Saha, C. Lundquist, and F. Gustafsson, "Marginalized adaptive particle filtering for nonlinear models with unknown time-varying noise parameters," *Automatica*, vol. 49, no. 6, pp. 1566–1575, 2013.

[23] A. Wills, T. B. Schön, F. Lindsten, and B. Ninness, "Estimation of linear systems using a Gibbs sampler," in *IFAC Symp. System Identification*, Brussels, Belgium, 2012.

[24] A. P. Dawid, "Some matrix-variate distribution theory: notational considerations and a Bayesian application," *Biometrika*, vol. 68, no. 1, pp. 265–274, 1981.

[25] 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.