Particle Gibbs with Ancestor Sampling for Identification of Tire-Friction Parameters

Berntorp, K.; Di Cairano, S.

TR2017-093 July 09, 2017

Abstract

Particle Gibbs with Ancestor Sampling (PGAS) is a particle Markov chain Monte Carlo method (PMCMC) for Bayesian inference and learning. PGAS conditions on a referencestate trajectory in the underlying particle filter using ancestor sampling. In this paper, we leverage PGAS for identification of cornering-stiffness parameters in road vehicles only using production-grade sensors. The cornering-stiffness parameters are essential for describing the motion of the vehicle. We show how PGAS can be adapted to efficiently learn the stiffness parameters by conditioning on the noise-input trajectory instead of the state trajectory. We verify on a three-minute long experimental test drive that our method correctly identifies the tire-stiffness parameters.

World Congress of the International Federation of Automatic Control (IFAC)

Particle Gibbs with Ancestor Sampling for Identification of Tire-Friction Parameters

Karl Berntorp [*] **Stefano Di Cairano** [*]

[*] *Mitsubishi Electric Research Laboratories, Cambridge, MA 02139*
USA (e-mail: karl.o.berntorp@ieee.org).

Abstract: Particle Gibbs with Ancestor Sampling (PGAS) is a particle Markov chain Monte Carlo method (PMCMC) for Bayesian inference and learning. PGAS conditions on a reference-state trajectory in the underlying particle filter using ancestor sampling. In this paper, we leverage PGAS for identification of cornering-stiffness parameters in road vehicles only using production-grade sensors. The cornering-stiffness parameters are essential for describing the motion of the vehicle. We show how PGAS can be adapted to efficiently learn the stiffness parameters by conditioning on the noise-input trajectory instead of the state trajectory. We verify on a three-minute long experimental test drive that our method correctly identifies the tire-stiffness parameters.

Keywords: Particle filter, Monte Carlo method, Friction estimation, System identification

1. INTRODUCTION

Sequential Monte-Carlo methods (SMCs) (Doucet and Johansen, 2009) are a set of numerical Bayesian-inference methods aimed at estimating the posterior distribution of the state trajectory $\boldsymbol{x}_{1:T}$, conditioned on the measurement trajectory $\boldsymbol{y}_{1:T}$ and the underlying dynamical model. Recently, SMCs have been combined with Markov Chain Monte Carlo methods (MCMCs, see (Robert and Casella, 2004)) for Bayesian inference and learning, resulting in a class of particle MCMC (PMCMC) methods (Andrieu et al., 2010). PMCMC relies on SMC to construct MCMC kernels. In the Particle Gibbs (PG) sampler, the Markov kernel is constructed by executing an SMC where one reference (state) trajectory is set a priori. To alleviate the inherent path-degeneracy problem, there has been extensions of PG including, for example, adding a backward sweep in the underlying SMC (Lindsten and Schön, 2013). Notably, PG with ancestor sampling (PGAS) constructs the Markov kernel by running a particle filter (PF), where, in each iteration, one of the particle (state) trajectories is conditioned using an additional ancestral sampling step (Lindsten et al., 2014). PMCMC and PGAS have thus far found applications in state smoothing (Svensson et al., 2015), system identification (Schön et al., 2015), and finance (Nonejad, 2015).

In this paper, we adapt PGAS for identification of tire-stiffness parameters. The tire-road contact is what mainly generates the forces that alter the motion of a ground vehicle, and the knowledge of variables related to the tire–road interaction is essential for advanced driver-assistance systems. During normal driving, that is, far from the vehicle handling limits, the tire–road model is commonly assumed to be a static, linear relationship between force and slip with the tire stiffness defined as the proportionality constant, but to reliably determine the tire stiffness is complicated even with this simplification. Knowledge of the tire-stiffness parameters are are necessary for accurately describing the motion of the vehicle (Di Cairano et al. (2013); Berntorp et al. (2014)). The research on tire-stiffness estimation is extensive. Linear recursive regression with different types of sensor setups is a common alternative (Gustafsson, 1997; Lundquist and Schön, 2009; Lee et al., 2015), but because of the complexity of the identification problem, linear approaches are not guaranteed to provide global convergence. We have previously reported on an approach for real-time estimation of the tire stiffness (Berntorp and Di Cairano, 2016), but real-time schemes will always be more or less dependent on a good initial guess and inherently gives nonperfect estimates. In this paper we focus on the offline identification part, which can, for example, be used for providing initial estimates for real-time schemes and therefore increases robustness of the complete estimation system. The experimental results repeated in this paper indicate that PGAS can indeed correctly identify the tire stiffness using only standard, production-viable sensors in passenger vehicles.

Sometimes conditioning on the state trajectory, as is usually done in PGAS, leads to poor performance. Degeneracy arises in a number of applications, such as tracking (Gustafsson et al., 2002) and state estimation (Berntorp and Di Cairano, 2016; Berntorp, 2016). Intuitively, the reason is tied to needing the inversion of the system model is needed to update the parameters based on the state trajectory. Instead, in this paper we condition on the *noise-input trajectory*, which is a numerically stable and therefore more appealing approach, in certain scenarios.

Preliminaries: We consider the model

$$\boldsymbol{x}_{k+1} \sim f_{\boldsymbol{\theta}}(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k, \boldsymbol{u}_k), \tag{1a}$$

$$\boldsymbol{y}_k \sim g_{\boldsymbol{\theta}}(\boldsymbol{y}_k|\boldsymbol{x}_k, \boldsymbol{u}_k), \tag{1b}$$

where $\sim$ means distributed according to, $\boldsymbol{u}_k$ is the known input, hereafter without loss of generalization assumed zero, at time index k, $f_{\boldsymbol{\theta}}(\cdot)$ and $g_{\boldsymbol{\theta}}(\cdot)$ are the dynamics and measurement model, respectively, modeled as probability density functions and parameterized by the unknown vector $\boldsymbol{\theta}$. For the ease of notation, we will sometimes write $\boldsymbol{f}_k$ for $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_k, \boldsymbol{u}_k)$.

In this paper we focus on a particular form of (1),

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k) + \boldsymbol{g}(\boldsymbol{x}_k)\boldsymbol{v}_k, \tag{2a}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k) + \boldsymbol{e}_k, \tag{2b}$$

where $\boldsymbol{v}$ is Gaussian distributed according to $\boldsymbol{v} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with unknown mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$, and where the process noise $\boldsymbol{v}_k$ and the measurement noise $\boldsymbol{e}_k$ can be dependent on each other. The task is to estimate the mean and covariance of the process noise based on a batch of measurements $\boldsymbol{y}_{1:T}$ and a state trajectory $\boldsymbol{x}_{1:T}$. Note that in general all involved functions in (2) are allowed to depend on $\boldsymbol{\theta}$.

The aim in Bayesian system identification is to model the unknown vector $\boldsymbol{\theta}$ as a random variable with some prior distribution, $\boldsymbol{\theta} \sim \pi(\boldsymbol{\theta})$, and learn the posterior $p(\boldsymbol{\theta}|\boldsymbol{y}_{1:T})$, that is, the posterior of the parameters conditioned on the observed data $\boldsymbol{y}_{1:T} = \{\boldsymbol{y}_1, \dots, \boldsymbol{y}_T\}$ from time index 1 to time index T. To this end, we can estimate the joint state and parameter posterior distribution $p(\boldsymbol{\theta}, \boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})$ and then exploit marginalization,

$$\begin{aligned} p(\boldsymbol{\theta}|\boldsymbol{y}_{1:T}) &= \int p(\boldsymbol{\theta}, \boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})\,\mathrm{d}\boldsymbol{x}_{1:T} \\ &= \int p(\boldsymbol{\theta}|\boldsymbol{x}_{1:T}, \boldsymbol{y}_{1:T}) p_{\boldsymbol{\theta}}(\boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})\,\mathrm{d}\boldsymbol{x}_{1:T}. \end{aligned} \tag{3}$$

Hence, to solve the Bayesian system identification problem, a feasible approach is to first compute the posterior of the parameters conditioned on the complete data $\{\boldsymbol{x}_{1:T}, \boldsymbol{y}_{1:T}\}$ and then marginalize out as in (3).

2. VEHICLE MODEL AND PROBLEM STATEMENT

We use a single-track model (Berntorp, 2014) for estimation. The states to estimate are the longitudinal and lateral velocity, and yaw rate, that is, $\boldsymbol{x} = [v^X \; v^Y \; \dot{\psi}]^T$. This model is commonly used for control in advanced driver-assistance systems (Di Cairano et al., 2013). In general, the longitudinal and lateral tire forces are nonlinearly dependent on the wheel slip κ and slip angle α. However, we assume normal driving conditions, meaning that the tire forces can be expressed as

$$F^x \approx C^x\kappa, \quad F^y \approx C^y\alpha, \tag{4}$$

where C^x and C^y are the longitudinal and lateral (cornering) stiffness, respectively. In addition, we assume small acceleration and deceleration, implying that $F^x \approx 0$. Inserting (4) into the equations of motion for the single-track model, which are straightforward to derive (Berntorp, 2014), gives

$$m(\dot{v}^Y + v^X\dot{\psi}) = C_f^y\alpha_f\cos(\delta) + C_r^y\alpha_r, \tag{5a}$$

$$I\ddot{\psi} = l_f C_f^y \alpha_f \cos(\delta) - l_r C_r^y \alpha_r, \tag{5b}$$

where v^X and v^Y are the longitudinal and lateral vehicle velocity, respectively, $\dot{\psi}$ is the yaw rate of the vehicle, δ is the wheel angle, $l_f + l_r$ is the wheel base, m is the vehicle mass, I is the inertia, and where subscripts f, r stand for front and right, respectively.

The slip angles are computed as

$$\alpha_f \approx \delta\frac{v^Y + l_f\dot{\psi}}{v^X}, \quad \alpha_r \approx \frac{l_r\psi \dot{-} v^Y}{v^X},$$

The longitudinal velocity v^X and steering angle δ are treated as known inputs. This is consistent with many navigation systems, where dead reckoning is used to decrease state dimension. In practice, v^X can be estimated using the wheel-speed sensors, transmission-shaft speed sensors, accelerometers, or a combination of them. We treat the tire-stiffness parameters as deviations from a nominal component,

$$C^y = C_n^y + \Delta C^y, \tag{6}$$

where C_n is the nominal value of the respective stiffness and ΔC^y is the unknown part. The disturbance vector

$$\boldsymbol{v} = \begin{bmatrix}\Delta C_f^y \; \Delta C_r^y\end{bmatrix}^{\mathrm{T}} \tag{7}$$

is modeled as a Gaussian random variable according to $\boldsymbol{v}_k \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with unknown mean and covariance. Inserting (6) into (5) and discretizing, results in

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k) + \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k)\boldsymbol{v}_k, \tag{8}$$

where $\boldsymbol{u}_k = [v^X \; \delta]^{\mathrm{T}}$. Thus, (8) corresponds to (2a) (or generally to (1a)). We measure the lateral acceleration a_m^Y and yaw rate $\dot{\psi}_m$, forming the measurement vector $\boldsymbol{y}_k = [a_m^Y \; \dot{\psi}_m]^{\mathrm{T}}$. An automotive-grade inertial sensor has a bias b, which needs to be modeled for any realistic implementation. We model the bias for the lateral acceleration and the yaw rate as a random walk,

$$\boldsymbol{b}_{k+1} = \boldsymbol{b}_k + \boldsymbol{v}_{b,k}, \tag{9}$$

where $\boldsymbol{v}_{b,k}$ is modeled as a zero-mean Gaussian with known covariance matrix $\boldsymbol{Q}$. The measurement model can be written as

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k) + \boldsymbol{b}_k + \boldsymbol{e}_k, \tag{10}$$

which corresponds to (2b) (or generally to (1b)). A complicating factor is that the noise sources $\boldsymbol{v}_k$ and $\boldsymbol{e}_k$ are dependent on each other. To see this, note that $a^Y = \dot{v}^Y + v^X\dot{\psi}$ can be extracted from (5a) by dividing with the vehicle mass. Hence, since we measure the lateral acceleration, the discretized version of (5) will appear in (10), and $\boldsymbol{e}_k$ can be decomposed as $\boldsymbol{e}_k = \bar{\boldsymbol{g}}(\boldsymbol{x}_k, \boldsymbol{u}_k)\boldsymbol{v}_k + \bar{\boldsymbol{e}}_k$ with $\bar{\boldsymbol{e}}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R})$. The mean $\bar{\boldsymbol{\mu}}$ and covariance $\bar{\boldsymbol{\Sigma}}$ of the joint Gaussian distribution of the process noise and the measurement noise can therefore be written as

$$\bar{\boldsymbol{\mu}} = \begin{bmatrix}\boldsymbol{\mu} \\ \bar{\boldsymbol{g}}_k\boldsymbol{\mu}\end{bmatrix}, \tag{11a}$$

$$\bar{\boldsymbol{\Sigma}} = \begin{bmatrix}\boldsymbol{\Sigma} & \boldsymbol{\Sigma}\bar{\boldsymbol{g}}_k^{\mathrm{T}} \\ \bar{\boldsymbol{g}}_k\boldsymbol{\Sigma} & \bar{\boldsymbol{g}}_k\boldsymbol{\Sigma}\bar{\boldsymbol{g}}_k^{\mathrm{T}} + \boldsymbol{R}\end{bmatrix}. \tag{11b}$$

Remark 1. The goal of this paper is to identify the cornering stiffness $\{C_f^y, C_r^y\}$ subject to vehicle model (8) and the measurement model (10), where the inertial sensors have time-varying bias (9). The cornering stiffness influences the vehicle state, which is only implicitly observed through the inertial sensors. The estimation quality of the vehicle state heavily affects the identification of the noise statistics, and vice versa, by the multiplicative relation between tire stiffness and state in (5), which also implies that the process noise is state dependent. Furthermore, because of the correlation between process and measurement noise, also the measurement noise is dependent on the tire stiffness. A further complicating factor is that because we rely on inertial sensors, the measurements will be biased.

The considered problem is therefore hard to solve and as pointed out in (Berntorp and Di Cairano, 2016), linear estimation techniques are likely to function well only in certain scenarios with specific settings.

2.1 Problem Formulation

We formulate the problem as identifying the parameter vector $\boldsymbol{\theta} = \{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}$ of the mean and covariance of the tire stiffness by formulating a Bayesian system identification problem, where we estimate the posterior $p(\boldsymbol{\theta}|\boldsymbol{y}_{1:T})$ of the parameters conditioned on the entire measurement history by leveraging (3). We tackle this problem by approximating the joint posterior $p(\boldsymbol{\theta}, \boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})$ with a PMCMC approach and then perform marginalization as in (3) to recover the stiffness estimates.

3. SEQUENTIAL MONTE CARLO AND MARKOV CHAIN MONTE CARLO

PFs approximate the posterior density $p_\theta(\boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})$ by a set of N weighted state trajectories as

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T}) \approx \sum_{i=1}^{N} w_T^i \delta_{\boldsymbol{x}_{1:T}^i}(\boldsymbol{x}_{1:T}), \tag{12}$$

where w_T^i is the importance weight of the ith trajectory $\boldsymbol{x}_{1:T}^i$ and $\delta(\cdot)$ is the Dirac function. The PF recursively estimates (12) by utilizing Bayes theorem through the equation

$$p_\theta(\boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T}) = \frac{g_\theta(\boldsymbol{y}_k|\boldsymbol{x}_k)p_\theta(\boldsymbol{x}_{1:T}|\boldsymbol{y}_{0:T-1})}{p_\theta(\boldsymbol{y}_T|\boldsymbol{y}_{0:T-1})}. \tag{13}$$

By introducing a *proposal density* for generating the samples $\{\boldsymbol{x}_T^i\}_{i=1}^N$,

$$\boldsymbol{x}_T \sim q_\theta(\boldsymbol{x}_T|\boldsymbol{x}_{T-1}, \boldsymbol{y}_T), \tag{14}$$

combining (12) and (13) leads to the importance weight

$$w_T^i = w_{T-1}^i \frac{g_\theta(\boldsymbol{y}_T|\boldsymbol{x}_T^i)f_\theta(\boldsymbol{x}_T^i|\boldsymbol{x}_{T-1}^i)}{q_\theta(\boldsymbol{x}_T^i|\boldsymbol{x}_{T-1}^i, \boldsymbol{y}_T)}. \tag{15}$$

In practice, PFs suffer from *path degeneracy*, which implies that (12) will be a poor approximation for any finite N and large T. This arises because of the (necessary) resampling step inherent in the PF. Resampling removes particles with low weights and replaces them with more likely particles and therefore diversity among the particles is lost. Hence, the PF estimate collapses for large T. PFs are therefore often combined with a backward sweep starting from the marginal density $p_\theta(\boldsymbol{x}_T|\boldsymbol{y}_{1:T})$, which typically can be well approximated by the PF without suffering from degeneracy. This is done by discarding the history $\boldsymbol{x}_{1:T-1}$. The idea is that first a PF is used to construct the marginal posterior at time index T, whereby a backward pass is performed that adjusts the state $\boldsymbol{x}_k$ with the more recent measurements $\boldsymbol{y}_{k+1}, \dots, \boldsymbol{y}_T$.

3.1 Particle Markov Chain Monte Carlo

MCMCs (Robert and Casella, 2004; Andrieu et al., 2010) can be used to sample from complicated distributions $\pi(\boldsymbol{\theta})$. The idea is to simulate a Markov chain that has π as stationary distribution. For instance, MCMC can be used to generate $\{\boldsymbol{\theta}(0), \boldsymbol{\theta}(1), \dots, \boldsymbol{\theta}(m)\}$ where $\boldsymbol{\theta}(m)$ depends on $\boldsymbol{\theta}(m-1)$ at each iteration, which for sufficiently large m are samples from π, that is, has π as stationary distribution. If the chain is *ergodic*, by the ergodic theorem (Robert and Casella, 2004) sample paths can be used to approximate expectations $\mathbb{E}_\pi$ with respect to π,

$$\frac{1}{M-k+1}\sum_{m=k}^{M} \phi(\boldsymbol{\theta}(m)) \to \mathbb{E}_\pi(\phi(\boldsymbol{\theta})), \quad M \to \infty, \tag{16}$$

for any test function ϕ where the first k samples in (16) belong to the burn-in phase, that is, the transient behavior, and are discarded.

MCMC methods are commonly used to sample from $p(\boldsymbol{\theta}, \boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})$ by alternately updating $\boldsymbol{x}_{1:T}$ given $\boldsymbol{\theta}$ and $\boldsymbol{\theta}$ given $\boldsymbol{x}_{1:T}$. An important step in the design of MCMC is constructing the Markov chain such that it converges to the distribution of interest. There are several approaches for achieving this. The Gibbs sampler is an MCMC method that estimates $p(\boldsymbol{\theta}, \boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})$ relying on the decomposition

$$p(\boldsymbol{\theta}, \boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T}) = p(\boldsymbol{\theta}|\boldsymbol{x}_{1:T}, \boldsymbol{y}_{1:T})p_\theta(\boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T}). \tag{17}$$

A possible benefit with using (17) is that it is often possible to sample from $p(\boldsymbol{\theta}|\boldsymbol{x}_{1:T}, \boldsymbol{y}_{1:T})$, and a proposal for $\boldsymbol{\theta}$, which can be tedious to design, is therefore avoided. Assuming that it is possible to sample a state trajectory from the smoothing density $p_{\boldsymbol{\theta}}(\boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})$, a procedure for approximating (17) is shown in Algorithm 1. Algorithm 1 produces the sequence of parameters and states $\{\boldsymbol{\theta}(m), \boldsymbol{x}_{1:T}(m)\}_{m=1}^M$, which forms a Markov chain. In the limit the simulated Markov chain has the density $p(\boldsymbol{\theta}, \boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})$ as stationary distribution, and under certain assumptions the Markov chain is consistent in the sense of (16). The smoothing density $p_\theta(\boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})$ can be intractable to sample from. In those cases we can replace exact sampling by leveraging SMC for constructing the samples on Line 3 in Algorithm 1, which still ensures convergence of the MCMC.

Algorithm 1 Bayesian learning of state-space models

```
1: Set $\boldsymbol{\theta}(0)$ and $\boldsymbol{x}_{1:T}(0)$ arbitrarily.
2: for $m \leftarrow 0$ to $M$ do
3:     Draw $\boldsymbol{x}_{1:T}(m+1) \sim p_{\boldsymbol{\theta}(m)}(\boldsymbol{x}_{1:T}(m)|\boldsymbol{y}_{1:T})$.
4:     Draw $\boldsymbol{\theta}(m+1) \sim p(\boldsymbol{\theta}|\boldsymbol{x}_{1:T}(m+1), \boldsymbol{y}_{1:T})$.
5: end for
```

The particle Gibbs with ancestor sampling (PGAS) is a PMCMC that estimates the smoothing density by a procedure similar to the standard PF, except for that the PF is conditioned on one prespecified *reference trajectory* $\boldsymbol{x}'_{1:T}$, which is retained throughout the procedure. In PGAS, at each time step $\boldsymbol{x}'_k$ is connected with one of the $N-1$ particles in the previous time step (i.e., one of the ancestors) by sampling a value for the *ancestor index* a_k^N with probability according to the respective importance weights $\{w_{k-1}^j\}_{j=1}^N$ of the particles at time $k-1$; in this way connections are made with the particles at the previous time index while still retaining a dependence on the reference trajectory. In (Lindsten et al., 2014) it is shown that Algorithm 1 using PGAS admits the joint distribution $p(\boldsymbol{\theta}, \boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})$ as stationary distribution. Furthermore, PGAS produces state trajectories that can be used as samples from the smoothing distribution $p_\theta(\boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})$ for any $N > 1$. PGAS takes a state trajectory $\boldsymbol{x}'_{1:T}$ and maps it onto another trajectory $\boldsymbol{x}_{1:T}$ and can therefore be viewed as a Markov kernel defined on the space of state trajectories.

4. PGAS BASED ON CONDITIONAL-INPUT PARTICLE FILTER FOR PARAMETER LEARNING

PGAS is based on a conditional particle filter with ancestral sampling (CPF-AS), where the conditioning is done with respect to the state trajectory. However, when PGAS is used for the purpose of learning $\boldsymbol{\theta}$, it can be inconvenient to condition on $\boldsymbol{x}_{1:T}$. One reason is that to generate the samples on Line 4 in Algorithm 1, inversion of parts of the dynamics is typically necessary. However, in practice this inversion can be avoided. To this end, this section formulates our modified PGAS for learning $\boldsymbol{\theta}$, which we apply to tire–road friction estimation.

Instead of only storing the ancestral states, resulting in the output state trajectory on Line 3 in Algorithm 1, we also store the noise-input sequence corresponding to the ancestral trajectories. The PGAS kernel is summarized in Algorithm 2. Then, the noise-input sequence is used to update the posterior density of the parameters on Line 4 in Algorithm 1. The only differences in Algorithm 2 compared with standard PGAS are in Lines 4, 7, 11, and 16, which are omitted in standard PGAS. However, this small modification has implications in practical applications of the framework. We will demonstrate the method in the next section. For an extensive discussion of particle ancestor sampling in nearly degenerate models and methods to alleviate this, see (Lindsten et al., 2015).

Algorithm 2 PGAS kernel

```
    Initialize: Draw $\{\boldsymbol{x}_1^i\}_{i=1}^{N-1} \sim p(\boldsymbol{x}_1)$ and set $\boldsymbol{x}_1^N = \boldsymbol{x}_1'$, $\{w_1^i\}_{i=1}^{N} = g_{\boldsymbol{\theta}}(\boldsymbol{y}_0|\boldsymbol{x}_0^i)$.
 1: for $k \leftarrow 2$ to $T$ do
 2:     for $i \leftarrow 1$ to $N-1$ do
 3:         Draw $a_k^i$ with $\mathbb{P}(a_k^i = j) \propto w_{k-1}^j$.
 4:         Draw $\{\boldsymbol{x}_k^i, \boldsymbol{v}_{k-1}^i\} \sim q_{\boldsymbol{\theta}}(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^{a_k^i}, \boldsymbol{y}_k)$.
 5:     end for
 6:     Set $\boldsymbol{x}_k^N = \boldsymbol{x}_k'$.
 7:     Set $\boldsymbol{v}_{k-1}^N = \boldsymbol{v}_{k-1}'$.
 8:     Draw $a_k^N$ with $\mathbb{P}(a_k^N = j) \propto w_{k-1}^j f_{\boldsymbol{\theta}}(\boldsymbol{x}_k'|\boldsymbol{x}_{k-1}^j)$.
 9:     for $i \leftarrow 1$ to $N$ do
10:         Set $\boldsymbol{x}_{1:k}^i = \{\boldsymbol{x}_{1:k}^{a_k^i}, \boldsymbol{x}_k^i\}$.
11:         Set $\boldsymbol{v}_{0:k-1}^i = \{\boldsymbol{v}_{0:k-2}^{a_k^i}, \boldsymbol{v}_{k-1}^i\}$.
12:         Set $w_k^i \propto g_{\boldsymbol{\theta}}(\boldsymbol{y}_k|\boldsymbol{x}_k^i) f_{\boldsymbol{\theta}}(\boldsymbol{x}_k^i|\boldsymbol{x}_{k-1}^i)/q_{\boldsymbol{\theta}}(\boldsymbol{x}_k^i|\boldsymbol{x}_{k-1}^i, \boldsymbol{y}_k)$.
13:     end for
14: end for
15: Draw $J$ with $\mathbb{P}(i = J) \propto w_T^i$.
16: Set $\boldsymbol{x}_{1:T}' = \boldsymbol{x}_{1:T}^J$, $\boldsymbol{v}_{0:T-1}' = \boldsymbol{v}_{0:T-1}^J$.
Output: $\{\boldsymbol{x}_{1:T}', \boldsymbol{v}_{0:T-1}'\}$
```

5. PGAS FOR ROAD-FRICTION ESTIMATION

Because the tire-stiffness parameters $\boldsymbol{\theta} = \{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}$ are modeled as Gaussian process noise, we can assign a conjugate prior (Murphy, 2007) to the prior $\pi(\boldsymbol{\theta})$ of the parameters. If a prior distribution belongs to the same family as the posterior distribution, the prior is conjugate to the likelihood. For Gaussian distributed data $\boldsymbol{v}$ we can use a Normal-inverse Wishart distribution as conjugate prior for the mean and covariance according to $\boldsymbol{\theta} \sim \mathrm{NiW}(\boldsymbol{\mu}_0, \lambda, \Psi, \nu)$, where $\{\boldsymbol{\mu}_0, \lambda, \Psi, \nu\}$ are the hyperparameters of the Normal-inverse Wishart. The resulting posterior distribution for the mean and covariance will also be a Normal-inverse Wishart,

$$\boldsymbol{\theta}|\boldsymbol{v} \sim \mathrm{NiW}(\boldsymbol{\mu}_T, \lambda_T, \Psi_T, \nu_T), \tag{18}$$

where the hyperparameters are (Murphy, 2007)

$$\begin{aligned} \boldsymbol{\mu}_T &= \frac{\lambda\boldsymbol{\mu}_0 + (T+1)\bar{\boldsymbol{v}}}{\lambda + T}, \\ \lambda_T &= \lambda + T, \\ \nu_T &= \nu + T, \\ \Psi_T &= \Psi + \boldsymbol{S} + \frac{\lambda T}{\lambda + t}(\bar{\boldsymbol{v}} - \boldsymbol{\mu}_0)^{\mathrm{T}}(\bar{\boldsymbol{v}} - \boldsymbol{\mu}_0), \end{aligned}$$

in which

$$\bar{\boldsymbol{v}} = \frac{1}{T}\sum_{k=1}^{T}\boldsymbol{v}_k, \quad \boldsymbol{S} = \sum_{k=1}^{T}(\boldsymbol{v}_k - \bar{\boldsymbol{v}})(\boldsymbol{v}_k - \bar{\boldsymbol{v}})^{\mathrm{T}}$$

are the mean and scatter matrix, respectively. Hence, samples from $p(\boldsymbol{\theta}|\boldsymbol{x}_{1:T}, \boldsymbol{y}_{1:T}) = p(\boldsymbol{\theta}|\boldsymbol{v}_{0:T-1})$ in Algorithm 1 are generated from (18).

In the underlying CPF, the total number of states to estimate consists of the vehicle state and the bias state. The CPF therefore targets the joint posterior density $p(\boldsymbol{x}_{1:k}, \boldsymbol{b}_k|\boldsymbol{y}_{1:k})$ at each time step k to create the reference trajectory necessary for estimating the tire-stiffness parameters. The joint posterior can be decomposed as

$$p(\boldsymbol{b}_k, \boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k}) = p(\boldsymbol{b}_k|\boldsymbol{x}_{1:k}, \boldsymbol{y}_{1:k})p(\boldsymbol{x}_{1:k}|\boldsymbol{y}_{1:k}). \tag{19}$$

Thus, the computation of the joint posterior can be done recursively by alternately estimating the vehicle state and the bias state.

Estimating the Vehicle State Choosing a proper proposal density (14) is important for reliable performance of any PF implementation, because the proposal alone determines how the particles are predicted. A suboptimal but rather common choice that is used in the bootstrap PF (Gordon et al., 1993) is to use the dynamics (1a) for propagating the particles. Because the inertial sensors introduce a dependence between the process noise and the measurement noise, the standard proposal needs to be modified to account for such dependence (Saha and Gustafsson, 2012). Since we model the unknown part of the stiffness parameters as a Gaussian random variable, this results in the Gaussian proposal

$$\begin{aligned} &q_\theta(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}, \boldsymbol{y}_{k-1}) = \\ &\mathcal{N}\Big(\boldsymbol{f}_k + \boldsymbol{g}_k\boldsymbol{\Sigma}\bar{\boldsymbol{g}}_k^{\mathrm{T}}(\bar{\boldsymbol{g}}_k\boldsymbol{\Sigma}\bar{\boldsymbol{g}}_k^{\mathrm{T}} + \boldsymbol{P}_k + \boldsymbol{R}_k)^{-1}(\boldsymbol{y}_k - \boldsymbol{h}_k - \boldsymbol{b}_k), \\ &\qquad \boldsymbol{g}_k(\boldsymbol{\Sigma} - \boldsymbol{\Sigma}\bar{\boldsymbol{g}}_k^{\mathrm{T}}(\bar{\boldsymbol{g}}_k\boldsymbol{\Sigma}\bar{\boldsymbol{g}}_k^{\mathrm{T}} + \boldsymbol{P}_k + \boldsymbol{R}_k)^{-1}\bar{\boldsymbol{g}}\boldsymbol{\Sigma})\boldsymbol{g}_k^{\mathrm{T}}\Big). \end{aligned} \tag{20}$$

Estimating the Bias The vehicle state only affects the bias state through the measurement equation (10), which is affine in $\boldsymbol{b}_k$ given the state trajectory $\boldsymbol{x}_{1:k}$. Furthermore, the time evolution of the bias is described by a Gaussian random walk (9), which is linear and independent from the state. Hence, given the state trajectory, the posterior of the bias is a Gaussian according to

$$p(\boldsymbol{b}_k|\boldsymbol{x}_{1:k}, \boldsymbol{y}_{1:k}) = \mathcal{N}(\hat{\boldsymbol{b}}_{k|k}, \boldsymbol{P}_{k|k}), \tag{21}$$

where the mean and covariance are computed with a Kalman filter conditioned on the state trajectory $\boldsymbol{x}_{1:T}$, resulting in the update equations

$$\hat{\boldsymbol{b}}_{k+1|k} = \hat{\boldsymbol{b}}_{k|k}, \quad \boldsymbol{P}_{k+1} = \boldsymbol{P}_k + \boldsymbol{Q}, \tag{22}$$

for the time update and

$$\begin{aligned} \hat{\boldsymbol{b}}_{k|k} &= \hat{\boldsymbol{b}}_{k|k-1} + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{h}_k - \bar{\boldsymbol{g}}_k\boldsymbol{\mu} - \hat{\boldsymbol{b}}_{k|k-1}), \\ \boldsymbol{P}_{k|k} &= \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k\boldsymbol{S}_k^{-1}\boldsymbol{K}_k^{\mathrm{T}}, \\ \boldsymbol{K}_k &= \boldsymbol{P}_{k|k-1}\boldsymbol{S}_k^{-1}, \\ \boldsymbol{S}_k &= (\boldsymbol{P}_{k|k-1} + \boldsymbol{R} + \bar{\boldsymbol{g}}_k\boldsymbol{\Sigma}\bar{\boldsymbol{g}}_k^{\mathrm{T}}). \end{aligned} \tag{23}$$

for the measurement update, where (22), (23) are taken per particle.

We summarize the resulting PGAS for this specific application in Algorithm 3, which replaces the generic Algorithm 2 on Line 3 in Algorithm 1.

Algorithm 3 PGAS kernel for Tire-Stiffness Identification

```
Initialize: Draw $\{\boldsymbol{x}_1^i\}_{i=1}^{N-1} \sim p(\boldsymbol{x}_1)$ and set $\boldsymbol{x}_1^N = \boldsymbol{x}'_1$, $\{w_1^i\}_{i=1}^N = g_{\boldsymbol{\theta}}(\boldsymbol{y}_0|\boldsymbol{x}_0^i)$.
for $k \leftarrow 2$ to $T$ do
    for $i \leftarrow 1$ to $N-1$ do
        Draw $a_k^i$ with $\mathbb{P}(a_k^i = j) \propto w_{k-1}^j$.
        Measurement update of bias ancestor $a_k^i$ using (23).
        Draw $\{\boldsymbol{x}_k^i, \boldsymbol{v}_{k-1}^i\} \sim q_{\boldsymbol{\theta}}(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^{a_k^i}, \boldsymbol{y}_{k-1})$ using (20).
        Predict bias statistics from ancestor $a_k^i$ using (22).
    end for
    Set $\boldsymbol{x}_k^N = \boldsymbol{x}'_k$.
    Set $\boldsymbol{v}_{k-1}^N = \boldsymbol{v}'_{k-1}$.
    Draw $a_k^N$ with $\mathbb{P}(a_k^N = j) \propto w_{k-1}^j f_{\boldsymbol{\theta}}(\boldsymbol{x}'_k|\boldsymbol{x}_{k-1}^j)$ using (8).
    for $i \leftarrow 1$ to $N$ do
        Set $\boldsymbol{x}_{1:k}^i = \{\boldsymbol{x}_{1:k}^{a_k^i}, \boldsymbol{x}_k^i\}$.
        Set $\boldsymbol{v}_{0:k-1}^i = \{\boldsymbol{v}_{0:k-2}^{a_k^i}, \boldsymbol{v}_{k-1}^i\}$.
        Set $w_k^i \propto g_{\boldsymbol{\theta}}(\boldsymbol{y}_k|\boldsymbol{x}_k^i) f_{\boldsymbol{\theta}}(\boldsymbol{x}_k^i|\boldsymbol{x}_{k-1}^i)/q_{\boldsymbol{\theta}}(\boldsymbol{x}_k^i|\boldsymbol{x}_{k-1}^i, \boldsymbol{y}_k)$ using (8), (10), and (20).
    end for
end for
Draw $J$ with $\mathbb{P}(i = J) \propto w_T^i$.
Set $\boldsymbol{x}'_{1:T} = \boldsymbol{x}_{1:T}^J$, $\boldsymbol{v}'_{0:T-1} = \boldsymbol{v}_{0:T-1}^J$.
Output: $\{\boldsymbol{x}'_{1:T}, \boldsymbol{v}'_{0:T-1}\}$
```

5.1 Experimental Results

We have used a mid-size SUV, equipped with state-of-the-art validation equipment, to gather data. The parameters of the vehicle model are extracted from data sheets and experimental validation. The vehicle is equipped with state-of-the-art sensors, only used to verify the algorithm, and the ground truth of the lateral stiffness parameters has previously been obtained by extensive vehicle experiments. The model (5) assumes knowledge of the front-wheel steering angle, which is not measured. However, the angle of the steering wheel, available from the CAN bus, is converted to a steering angle of the front wheel using a constant gear ratio. The stiffness values are initialized to 50% of the true values. Note that we also tried the original formulation of PGAS, but the convergence was not reliable enough and the results are therefore omitted.

The data set consists of normal driving on a regular dry asphalt road and is about 300 seconds long. We stress that this data set is collected from a period of regular driving on a standard two-lane road and was not gathered for the purpose of this experiment. The road requires only light steering, which reduces observability, and it contains nonzero inclination and bank angles, which are not explicitly accounted for in the current implementation. Thus, the dataset also tests how robust the algorithm is to these unmodeled effects.

Fig. 1 displays the estimated mean and standard deviation of the cornering stiffness for both wheel axles when executing Algorithm 1 using $M = 5000$ iterations. The underlying CPF in Algorithm 3 uses $N = 400$ particles. The measurements are gathered at 100 Hz, which is also the time step used in the discretization of the dynamics. Thus, the CPF executes for $T = 30\ 000$ time steps. The tire-stiffness estimates converge very close to the true values and the estimated standard deviation is about 8% of the true value. It is interesting to compare with stiffness values obtained on different surfaces. It is difficult to give exact thresholds on how much the stiffness values differ between different surfaces, because they depend on a number of things, such as the specific tire, the evenness of the actual road stretch, and more. However, it is still possible to deduct that the stiffness values between snow and dry asphalt differ about a factor of two, and the corresponding differences between dry and wet asphalt are in the order of 20% (Svendenius, 2007). Hence, it is obvious from Fig. 1 that the algorithm can distinguish between dry asphalt and snow on this particular data set. Furthermore, it is likely that it can distinguish dry and wet asphalt with high certainty, although more tests have to be made to verify this statement.

In Fig. 2 we show the resulting posterior estimate (i.e., $p(\boldsymbol{\theta}|\boldsymbol{y}_{1:T})$) of the tire stiffness, excluding the burn-in phase, normalized with respect to the true values. Note that although the ground-truth values have been calibrated using high-precision instruments, also the ground truth has some inherent uncertainty because of, for example, slight variations in tire load, tire pressure, and temperature, between the time of calibration and the time of collection of the considered data set. The posterior means are the red dashed vertical lines. In the upper plot, the error compared with the true value is about 3%. The rear lateral stiffness estimate deviates less than 1%, which is remarkable considering that the method only uses production sensors and that the data set has not been gathered for the purpose of identifying the tire stiffness.

6. CONCLUSION

The main contribution of this paper is adaptation and implementation of the particle Gibbs with ancestor sampling for identification of the cornering stiffness parameter that are necessary in many advanced vehicle-control and estimation applications. The method only uses inertial and wheel-speed sensors, which are typically installed in production vehicles. We explained how to implement the underlying conditional particle filter for this application. The method was verified on a three minute long data set taken from a test drive on dry asphalt. The experimental results on dry asphalt show that the method leads to stiffness estimates that deviate less than 1% from the true values after the transient phase. It is future work to fully evaluate the algorithm on more data sets and vehicle configurations, based on these highly encouraging initial results.

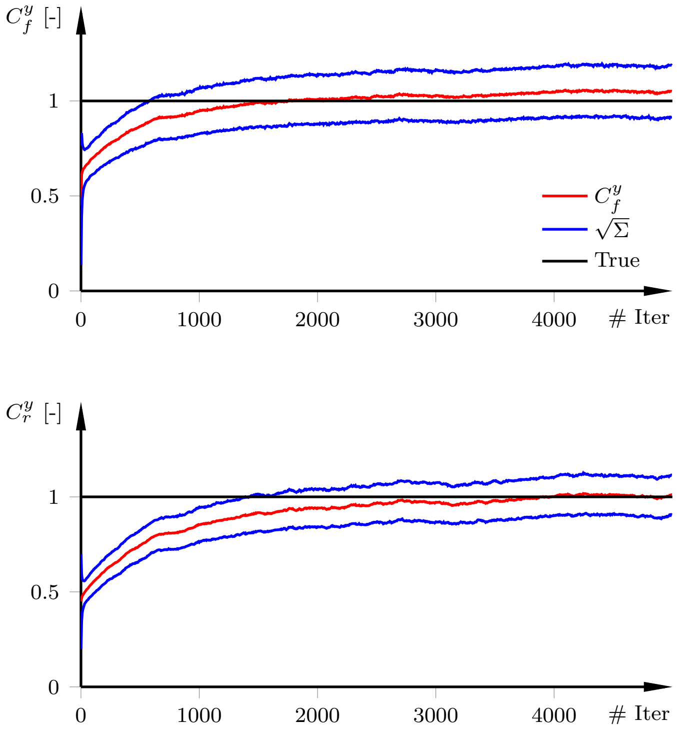

Fig. 1. Identified mean (red) and standard deviation (blue) of the cornering stiffness, respectively, for the front (upper two plots) and rear axle. True values are shown in black. The scales are normalized with respect to the ground-truth value because of confidentiality.

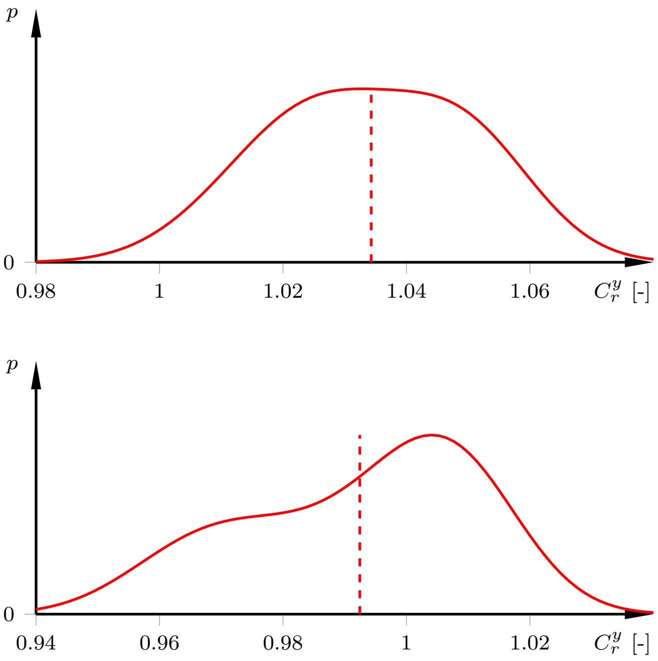

Fig. 2. Posterior estimates ($p(\boldsymbol{\theta}|\boldsymbol{y}_{1:T})$) for the front (upper plot) and rear lateral stiffness. The red dashed vertical lines signify the posterior means. The scales are normalized with respect to the ground-truth value because of confidentiality.

REFERENCES

Andrieu, C., Doucet, A., and Holenstein, R. (2010). Particle markov chain monte carlo methods. *J. Royal Statistical Society: Series B (Statistical Methodology)*, 72(3), 269–342.

Berntorp, K. (2016). Joint wheel-slip and vehicle-motion estimation based on inertial, GPS, and wheel-speed sensors. *IEEE Trans. Control Syst. Technol.*, 24(3), 1020–1027.

Berntorp, K. (2014). *Particle Filtering and Optimal Control for Vehicles and Robots*. Ph.D. thesis, Dept. Automatic Control, Lund University, Sweden.

Berntorp, K. and Di Cairano, S. (2016). Tire-stiffness estimation by marginalized adaptive particle filter. In *55th IEEE Int. Conf. Decision and Control*. Las Vegas, NV. To appear.

Berntorp, K., Olofsson, B., Lundahl, K., and Nielsen, L. (2014). Models and methodology for optimal trajectory generation in safety-critical road–vehicle manoeuvres. *Veh. Syst. Dyn.*, 52(10), 1304–1332.

Di Cairano, S., Tseng, H., Bernardini, D., and Bemporad, A. (2013). Vehicle yaw stability control by coordinated active front steering and differential braking in the tire sideslip angles domain. *IEEE Trans. Control Syst. Technol.*, 21(4), 1236–1248.

Doucet, A. and Johansen, A.M. (2009). A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of nonlinear filtering*, 12(656-704), 3.

Gordon, N.J., Salmond, D.J., and Smith, A.F.M. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *Radar and Signal Processing, IEE Proc. F*, 140(2), 107–113.

Gustafsson, F., Gunnarsson, F., Bergman, N., Forssell, U., Jansson, J., Karlsson, R., and Nordlund, P.J. (2002). Particle filters for positioning, navigation, and tracking. *IEEE Trans. Signal Process.*, 50(2), 425–437.

Gustafsson, F. (1997). Slip-based tire-road friction estimation. *Automatica*, 33, 1087–1099.

Lee, S., Nakano, K., and Ohori, M. (2015). On-board identification of tyre cornering stiffness using dual Kalman filter and GPS. *Veh. Syst. Dyn.*, 53(4), 437–448.

Lindsten, F., Bunch, P., Singh, S.S., and Schön, T.B. (2015). Particle ancestor sampling for near-degenerate or intractable state transition models. *ArXiv e-prints*.

Lindsten, F., Jordan, M.I., and Schön, T.B. (2014). Particle Gibbs with ancestor sampling. *J. Machine Learning Res.*, 15(1), 2145–2184.

Lindsten, F. and Schön, T.B. (2013). Backward simulation methods for Monte Carlo statistical inference. *Foundations and Trends in Machine Learning*, 6(1), 1–143.

Lundquist, C. and Schön, T.B. (2009). Recursive identification of cornering stiffness parameters for an enhanced single track model. In *15th IFAC Symp. System Identification*. Saint-Malo, France.

Murphy, K.P. (2007). Conjugate Bayesian analysis of the Gaussian distribution. Technical report, UBC.

Nonejad, N. (2015). Particle Gibbs with ancestor sampling for stochastic volatility models with: Heavy tails, in mean effects, leverage, serial dependence and structural breaks. *Studies in Nonlinear Dynamics and Econometrics*, 19(5), 561–584.

Robert, C.P. and Casella, G. (2004). *Monte Carlo Statistical Method*. Springer, Secaucus, NJ, USA.

Saha, S. and Gustafsson, F. (2012). Particle filtering with dependent noise processes. *IEEE Trans. Signal Process.*, 60(9), 4497–4508.

Schön, T.B., Lindsten, F., Dahlin, J., Wågberg, J., Naesseth, C.A., Svensson, A., and Dai, L. (2015). Sequential Monte Carlo methods for system identification. In *IFAC Symp. System Identification*. Bejing, China.

Svendenius, J. (2007). *Tire modeling and Friction Estimation*. Ph.D. thesis, Dept. Automatic Control, Lund University, Sweden.

Svensson, A., Schön, T.B., and Kok, M. (2015). Nonlinear state space smoothing using the conditional particle filter. In *IFAC Symp. System Identification*. Bejing, China.