A semiparametric Bayesian approach to Wiener system identification

Fredrik Lindsten * **Thomas B. Schön** * **Michael I. Jordan** **

* *Division of Automatic Control, Linköping University, Linköping, Sweden (e-mail: {lindsten,schon}@isy.liu.se)*
** *Departments of EECS and Statistics, University of California, Berkeley, USA (e-mail: jordan@eecs.berkeley.edu)*

Abstract: We consider a semiparametric, i.e. a mixed parametric/nonparametric, model of a Wiener system. We use a state-space model for the linear dynamical system and a nonparametric Gaussian process (GP) model for the static nonlinearity. The GP model is a flexible model that can describe different types of nonlinearities while avoiding making strong assumptions such as monotonicity. We derive an inferential method based on recent advances in Monte Carlo statistical methods, known as Particle Markov Chain Monte Carlo (PMCMC). The idea underlying PMCMC is to use a particle filter (PF) to generate a sample state trajectory in a Markov chain Monte Carlo sampler. We use a recently proposed PMCMC sampler, denoted particle Gibbs with backward simulation, which has been shown to be efficient even when we use very few particles in the PF. The resulting method is used in a simulation study to identify two different Wiener systems with non-invertible nonlinearities.

1. INTRODUCTION

Block-oriented nonlinear systems are a family of nonlinear dynamical systems which have attracted significant attention in the system identification community, see e.g. [Giri and Bai, 2010]. These systems consist of interconnected linear dynamical systems and static nonlinearities. The most well-known members of this family are the Hammerstein (static nonlinearity followed by a linear dynamical system) and the Wiener (linear dynamical system followed by a static nonlinearity) systems, introduced by Hammerstein [1930] and Wiener [1966], respectively.

We are concerned here with the problem of "blind identification" of a Wiener system; i.e., the case when the identification is carried out in the absence of a known input signal. In other words, we wish to identify a model of a Wiener system based on the information present in the measurements $y_{1:T} \triangleq \{y_t\}_{t=1}^T$; see Figure 1. This problem has attracted significant interest, see e.g. [Vanbeylan et al., 2009, Abed-Meraim et al., 1997, Bai, 2002, Wills et al., 2011]. However, it should be noted that the proposed method can be generalised straightforwardly to the case in which the system is excited by a known input signal as well.

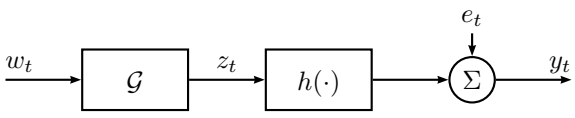

Fig. 1. A blind Wiener system, consisting of a linear system $\mathcal{G}$ followed by a static nonlinearity $h(\cdot)$. The system noise w_t and the measurement noise e_t are both unmeasurable.

⋆ This work was supported by: the project Calibrating Nonlinear Dynamical Models (Contract number: 621-2010-5876) funded by the Swedish Research Council and CADICS, a Linneaus Center also funded by the Swedish Research Council.

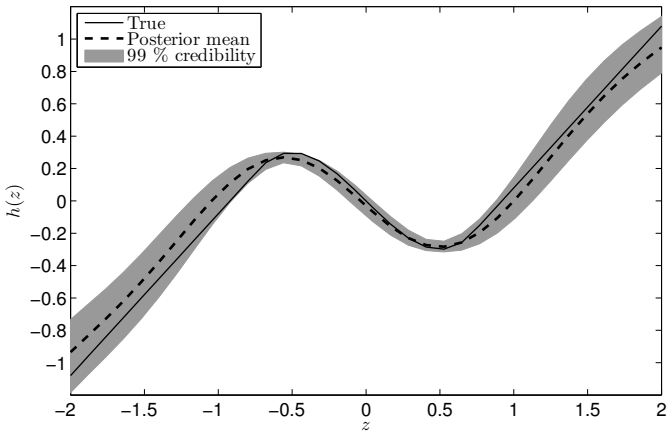

Fig. 2. Nonlinear mapping (non-monotone), the estimated posterior mean and 99 % credibility interval.

We consider a semiparametric (i.e., a mixed parametric/nonparametric) model of a Wiener system. We use a state-space model for the linear dynamical system and a nonparametric Gaussian process (GP) model for the static nonlinearity. We take a Bayesian approach, modeling the unknown parameters of the model as random variables. The objective in this work is then to provide a method for computing $p(\theta \mid y_{1:T})$, the posterior probability density function (PDF) of the unknown parameters θ given the measurements $y_{1:T}$. The posterior PDF does not allow for a closed form solution and we will make use of a Markov Chain Monte Carlo (MCMC) method (see e.g. [Robert and Casella, 2004] for a general introduction) to compute an approximation of $p(\theta \mid y_{1:T})$. More specifically, we employ the recently proposed particle MCMC (PMCMC) framework by Andrieu et al. [2010]. The basic idea underlying PMCMC is to use a particle filter (PF) to generate a sample state trajectory, which is then used as a component of an MCMC sampler. Here, we use a PMCMC sampler denoted particle Gibbs with backward simulation

(PG-BSi). This was originally proposed by Whiteley [2010] and further explored by Lindsten and Schön [2012], who also illustrated its efficiency even when we use very few particles in the underlying PF.

Due to the nonparametric nature of the GP, the model proposed in this work is flexible and can be used for a wide range of nonlinear mappings. Furthermore, the inferential method, which is based on PG-BSi, does not impose strong assumptions such as invertibility or monotonicity of the nonlinear mapping. We illustrate the type of results that we are able to obtain with the proposed metod in Figure 2. This figure shows the nonparametric estimate of a non-monotonic nonlinearity. Here, the preceding linear system was a fourth-order oscillatory system. The full experimental details are given in Section 5.2.

To the best of our knowledge this is the first time the posterior PDF $p(\theta \mid y_{1:T})$ is computed for the blind Wiener problem. There are maximum likelihood approaches for solving the blind Wiener problem available [Vanbeylan et al., 2009, Wills et al., 2011], where the former makes the restrictive assumption that the nonlinearity is invertible. Pillonetto and Chiuso [2009] have provided an interesting nonparametric approach for Wiener identification using GPs. However, they are only concerned with finding maximum a posteriori point estimates and do not compute the full posterior PDF.

2. A SEMIPARAMETRIC BAYESIAN MODEL

In this work, we consider a semiparametric model of the (blind) Wiener system. The linear dynamical system is modeled using a (parametric) state-space representation, and we use a nonparametric Gaussian process (GP) model for the static nonlinearity. The model can be described in state-space form as

$$x_{t+1} = Ax_t + w_t, \qquad w_t \sim \mathcal{N}(0, Q), \tag{1a}$$
$$z_t = Cx_t, \tag{1b}$$
$$y_t = h(z_t) + e_t, \qquad e_t \sim \mathcal{N}(0, r). \tag{1c}$$

The linear system is assumed to be observable. Hence, we can, without loss of generality, fix the matrix C according to $C = (1\ 0\ \cdots\ 0)$. Then, the unknown quantities of the model are the parameters $\eta \triangleq \{A, Q, r\}$ as well as the nonlinear mapping $h(\cdot)$.

We take a Bayesian approach and model the parameters of the model as random variables. We place a matrix normal, inverse Wishart (MNIW) prior on the pair $\{A, Q\}$, $p(A, Q) = p(A \mid Q)p(Q)$ where,

$$p(A \mid Q) = \mathcal{MN}(A; M, Q, L), \tag{2a}$$
$$p(Q) = \mathcal{IW}(Q; n_0, S_0). \tag{2b}$$

Here $\mathcal{MN}(A; M, V, L)$ is a matrix normal density with mean matrix M and left and right covariances L^{-1} and V, respectively; $\mathcal{IW}(\Sigma; n, S)$ is an inverse Wishart density with n degrees of freedom and scale matrix S. The MNIW prior is conjugate to a linear Gaussian model such as (1a) and is a standard choice in Bayesian statistics (see e.g. West and Harrison [1997]). For suitably chosen hyperparameters (i.e. M, L, n_0 and S_0), the effects of this prior on the posterior density will be minor. For a discussion on how to choose the hyperparameters, see Appendix B. Similarly, we put an inverse Wishart (IW) prior on r (the univariate IW distribution is also known as inverse Gamma), according to,

$$p(r) = \mathcal{IW}(r; m_0, R_0). \tag{3}$$

For the nonlinear mapping we apply a nonparametric model by placing a GP prior on h,

$$h(\cdot) \sim \mathcal{GP}(m(z), k(z, z')). \tag{4}$$

A GP is a probability distribution over functions, which suggests its use as a nonparametric prior distribution in Bayesian statistics. See Rasmussen and Williams [2006] for a thorough introduction to GPs. The GP is governed by a mean function m and a covariance function (also referred to as a kernel) k. Here, we use $m(z) = z$, i.e. the prior is that no nonlinearity is present. The kernel is taken as squared exponential,

$$k(z, z') = \alpha \exp(-0.5(z - z')^2/\ell^2), \tag{5}$$

with amplitude α and length scale ℓ. However, both the mean function and covariance kernel may be chosen differently (though $m(z) = z$ seems to be a sensible choice). Note that, due to the nonparametric nature of the GP, the proposed model is flexible and can describe a wide range of nonlinear mappings. We do not assume any specific form of h. However, the GP using a squared exponential covariance kernel is a smoothness prior. Hence, our model will favor smooth functions h. Still, as we shall see in Section 5, the proposed method can perform well even when the true nonlinearity is non-differentiable.

3. INFERENCE VIA PARTICLE GIBBS SAMPLING

Assume that we have observed a sequence of measurements $y_{1:T} \triangleq \{y_1, \ldots, y_T\}$. The task at hand is to identify the unknown quantities of the model, i.e. the parameters η as well as the nonlinear mapping $h(\cdot)$. Let us introduce the augmented parameter $\theta \triangleq \{\eta, h(\cdot)\} \in \mathcal{S} \times \mathbb{F}$, where $\mathcal{S}$ is a finite-dimensional space (containing η) and $\mathbb{F}$ is an appropriate function space. We then seek the posterior density of the parameter θ given the observations $y_{1:T}$, or more generally the joint posterior density of the parameter and the system states $x_{1:T}$, i.e.

$$p(\theta, x_{1:T} \mid y_{1:T}) = p(x_{1:T} \mid \theta, y_{1:T})p(\theta \mid y_{1:T}). \tag{6}$$

Note that we use the term "parameter" to refer to θ, which includes also the nonparametric part of the model, h.

This posterior density is intractable and we shall make use of an MCMC sampler to address the inference problem. Consider η, $h(\cdot)$ and $x_{1:T}$ as three (collections of) variables of the model. We then suggest to use a three-step Gibbs sampler, targeting the density (6), which iterates the following three steps,

$$\text{Draw } \eta^\star \mid h, x_{1:T} \sim p(\eta \mid h, x_{1:T}, y_{1:T}); \tag{7a}$$
$$\text{Draw } h^\star \mid \eta^\star, x_{1:T} \sim p(h \mid \eta^\star, x_{1:T}, y_{1:T}); \tag{7b}$$
$$\text{Draw } x^\star_{1:T} \mid \theta^\star = \{\eta^\star, h^\star\} \sim p(x_{1:T} \mid \theta^\star, y_{1:T}). \tag{7c}$$

The reason for considering the split according to η, $h(\cdot)$ and $x_{1:T}$ is that the posterior parameter distributions appearing in (7a) and (7b) then will be available in closed form. Deriving these posterior densities will be the topic of Section 4.

Unfortunately, step (7c) of this Gibbs sweep is still intractable, since the joint smoothing density $p(x_{1:T} \mid \theta, y_{1:T})$ is not available in closed form for the model (1). In other words, the state inference problem is intractable, even if we fix the parameters of the model, due to the presence of the nonlinearity. However, it is possible to circumvent this problem by employing a powerful statistical inference tool, recently introduced by Andrieu et al. [2010], known as particle MCMC (PMCMC).

A thorough treatment of PMCMC is well beyond the scope of this paper and we refer the interested reader to [Andrieu et al., 2010, Lindsten and Schön, 2012]. However, in the remainder of this section we briefly introduce the particular PMCMC method that we have employed.

The basic idea behind PMCMC is to use a particle filter (PF) as a proposal kernel in an MCMC sampler. In step (7c) of the "idealised" Gibbs sampler outlined above, we wish to sample a state trajectory from the joint smoothing density, for a fixed parameter $\theta^\star$. This density is not available in closed form, but we can approximate it using a PF. Hence, consider the following sampling strategy. We parameterise the model with $\theta^\star$ and apply a PF to the data $y_{1:T}$. The PF will generate N particle trajectories with corresponding importance weights, $\{x^i_{1:T}, w^i_T\}_{i=1}^N$, which can be seen as a weighted sample from the joint smoothing density $p(x_{1:T} \mid \theta^\star, y_{1:T})$. Hence, if we sample among these trajectories, i.e. we choose $x^i_{1:T}$ with probability w^i_T, this will be an *approximate* realisation from $p(x_{1:T} \mid \theta^\star, y_{1:T})$.

Now, if we simply replace step (7c) of the idealised Gibbs sampler with the sampling strategy outlined above, the approximative nature of the PF will cause the Gibbs sampler to converge to the wrong distribution. However, what was shown by Andrieu et al. [2010] is that it is possible to *exactly* compensate for these approximations, by making a slight modification to the PF, leading to a similar approach called the conditional PF (CPF).

Since the introduction of PMCMC techniques by Andrieu et al. [2010], several contributions have been made, which make the methods even more appealing. In this work, we have employed the particle Gibbs with backward simulation (PG-BSi) sampler [Whiteley, 2010, Lindsten and Schön, 2012]. This method differs from the original particle Gibbs sampler, in that a backward simulator (see Godsill et al. [2004], Douc et al. [2011]) is used to generate a sample trajectory. By doing so, it is possible to mitigate the well known degeneracy problem, which otherwise deteriorates the PF. Lindsten and Schön [2012] show that backward simulation can increase the mixing of the PMCMC sampler significantly, especially when we use few particles in the underlying CPF. It is shown that the PG-BSi sampler can function properly even with extremely few particles. In this work, the PG-BSi sampler is used as a component in the proposed identification method, and in the examples considered in Section 5 we employ the PG-BSi sampler using only $N = 5$ particles, with good results.

4. POSTERIOR PARAMETER DISTRIBUTIONS

We now turn our attention to steps (7a)–(7b) of the Gibbs sampler. That is, we assume that a fixed state trajectory $x_{1:T}$ is given and consider the problem of sampling from the posterior parameter distributions. Conditioned on $x_{1:T}$, the variables $\{A, Q\}$ are independent of $\{h(\cdot), r\}$. Hence, the densities appearing in (7a) and (7b) can be written as

$$p(\eta \mid h, x_{1:T}, y_{1:T}) = p(A, Q \mid x_{1:T}, y_{1:T}) p(r \mid h, x_{1:T}, y_{1:T}), \tag{8a}$$

$$p(h \mid \eta, x_{1:T}, y_{1:T}) = p(h \mid r, x_{1:T}, y_{1:T}). \tag{8b}$$

Sampling from (8a) can thus be split into two decoupled steps. In the three subsequent sections, we derive the expressions for the three density functions appearing on the right hand sides of (8).

4.1 Posterior of A and Q

From the model (1) we have that $p(A, Q \mid x_{1:T}, y_{1:T}) = p(A, Q \mid x_{1:T})$. Let $X = [x_2 \ \dots \ x_T]$, $\bar{X} = [x_1 \ \dots \ x_{T-1}]$ and $W = [w_1 \ \dots \ w_{T-1}]$. It then follows from (1a) that the likelihood $p(x_{1:T} \mid A, Q)$ can be described in terms of the relation

$$X = A\bar{X} + W. \tag{9}$$

The prior (2) is conjugate to this likelihood model and it follows (see e.g. [West and Harrison, 1997]) that the posterior parameter distribution is MNIW and is given by,

$$\begin{aligned} p(A, Q \mid x_{1:T}) = \mathcal{MN}(A; S_{X\bar{X}} S_{\bar{X}\bar{X}}^{-1}, Q, S_{\bar{X}\bar{X}}) \\ \times \mathcal{IW}(Q; T - 1 + n_0, S_{X|\bar{X}} + S_0), \end{aligned} \tag{10a}$$

with

$$S_{\bar{X}\bar{X}} = \bar{X}\bar{X}^{\mathsf{T}} + L, \tag{10b}$$

$$S_{X\bar{X}} = X\bar{X}^{\mathsf{T}} + ML, \tag{10c}$$

$$S_{XX} = XX^{\mathsf{T}} + MLM^{\mathsf{T}}, \tag{10d}$$

$$S_{X|\bar{X}} = S_{XX} - S_{X\bar{X}} S_{\bar{X}\bar{X}}^{-1} S_{X\bar{X}}^{\mathsf{T}}. \tag{10e}$$

Hence, we can sample from the posterior (10) by first sampling Q from an IW distribution, and thereafter sample A from an MN distribution.

4.2 Posterior of r

For fixed $x_{1:T}$ and $h(\cdot)$, let $\mathbf{h} = (h(Cx_1) \ \cdots \ h(Cx_T))^{\mathsf{T}}$ and $\mathbf{y} = (y_1 \ \cdots \ y_T)^{\mathsf{T}}$ be the vectors of function outputs and observations, respectively. Furthermore, let $\mathbf{e} = (e_1 \ \cdots \ e_T)^{\mathsf{T}}$. It then follows from (1c) that the likelihood $p(y_{1:T} \mid r, h, x_{1:T})$ can be described in terms of the relation $\mathbf{y} = \mathbf{h} + \mathbf{e}$. The prior $p(r \mid h, x_{1:T}) = p(r)$ given in (3) is conjugate to this likelihood model and it follows that the posterior parameter distribution is IW and is given by,

$$p(r \mid h, x_{1:T}, y_{1:T}) = \mathcal{IW}(r; T + m_0, S_r + R_0), \tag{11}$$

with $S_r = (\mathbf{y} - \mathbf{h})^{\mathsf{T}}(\mathbf{y} - \mathbf{h})$.

4.3 Posterior of $h(\cdot)$

The GP prior (4) is conjugate to the likelihood model given by (1c). Hence, the posterior distribution of $h(\cdot)$ given r, $x_{1:T}$ and $y_{1:T}$ is a GP. Sampling from this posterior distribution thus involves drawing a sample path from the posterior stochastic process. When it comes to implementing a Gibbs sampler containing such a GP posterior, a problem that we need to address is how to represent this sample path. Since the index set $\mathbb{R}$ is uncountable, we can clearly not compute the value of the sample path at every point in the index set.

Here, we present two alternative approaches. The first, and most proper, solution is to sample from the GP whenever an evaluation of the function h is needed in the algorithm. This will be done for N query points for each time $t = 1, \dots, T$, where N is the number of particles used in the PG-BSi sampler (see Section 3). Hence, using this approach we need to sample sequentially from the posterior GP. In Appendix A we discuss how this can be done in an efficient way, based on a recursion of the Cholesky factor of the posterior covariance matrix.

The second alternative is a simpler approach, which is to evaluate the GP on a fixed grid of points. This is done

once for each iteration of the MCMC sampler, prior to the particle filtering. When evaluating the function h in the PF, we do a simple linear interpolation. The grid is chosen in such a way that (with probability close to 1) we never have to evaluate the function outside the grid. This is possible since we fix the scale of the input to the function, as described in Section 4.4. This approximate solution is the approach that we have employed in the numerical examples presented in Section 5.

In either approach, let $\mathbf{z}_\star = \begin{pmatrix} z^{(1)} & \dots & z^{(M)} \end{pmatrix}^\mathsf{T}$ be the points for which we wish to evaluate the GP (i.e., these can either be random points generated in the PF or some fixed grid-points). Furthermore, let $\mathbf{h}_\star = \begin{pmatrix} h(z^{(1)}) & \dots & h(z^{(M)}) \end{pmatrix}^\mathsf{T}$. It then follows (see [Rasmussen and Williams, 2006, Section 2.2]) that the posterior distribution of $\mathbf{h}_\star$ is given by,

$$p(\mathbf{h}_\star \mid r, x_{1:T}, y_{1:T}) = \mathcal{N}\left(\mathbf{h}_\star; \mu_\star, \Sigma_\star\right), \tag{12a}$$

where

$$\mu_\star = \mathbf{m}_\star + K_\star^\mathsf{T}(K + rI_T)^{-1}(\mathbf{y} - \mathbf{m}), \tag{12b}$$

$$\Sigma_\star = K_{\star\star} - K_\star^\mathsf{T}(K + rI_T)^{-1}K_\star. \tag{12c}$$

Here, I_d is a $d \times d$ identity matrix and we have introduced the notation

$$\mathbf{m}_\star = \begin{pmatrix} m(z^{(1)}) & \cdots & m(z^{(M)}) \end{pmatrix}^\mathsf{T}, \tag{13a}$$

$$\mathbf{m} = \begin{pmatrix} m(z_1) & \cdots & m(z_T) \end{pmatrix}^\mathsf{T}. \tag{13b}$$

Furthermore, the matrices K, $K_\star$ and $K_{\star\star}$ have elements given by,

$$[K]_{ij} = k(z_i, z_j), \qquad i, j = 1, \dots, T, \tag{13c}$$

$$[K_\star]_{ij} = k(z_i, z^{(j)}), \qquad i = 1, \dots, T,\ j = 1, \dots, M, \tag{13d}$$

$$[K_{\star\star}]_{ij} = k(z^{(i)}, z^{(j)}), \quad i, j = 1, \dots, M. \tag{13e}$$

Using the expressions above, we can generate a sample of $\mathbf{h}_\star$ from the posterior distribution (12). To obtain a numerically robust method, it is recommended to make use of a Cholesky factorisation to compute the posterior mean and covariance in (12); see Appendix A.

It should be noted that the computational complexity of evaluating and sampling from a posterior GP is cubic in the number of query points as well as the number of data points, i.e. of order $O(M^3 + T^3)$. If the GP is evaluated within the PF, i.e. according to the first alternative outlined above, $M = NT$. If we instead use a fixed grid, M is the number of grid points. In either case, the cost of sampling from the GP can be prohibitive, especially if T is large. However, there exist several methods in the literature, dedicated to enabling GP regression for large data sets, e.g. based on low-rank approximations; see [Rasmussen and Williams, 2006, Chapter 8] and the references therein. In this work we have not resorted to such techniques.

4.4 A scale ambiguity

Consider the model (1) and assume that we make a change of variables $\tilde{x}_t = cx_t$ and $\tilde{z}_t = cz_t$ for some positive constant c. An equivalent model to (1) is then given by

$$\tilde{x}_{t+1} = A\tilde{x}_t + \tilde{w}_t, \qquad \tilde{w}_t \sim \mathcal{N}(0, \tilde{Q}), \tag{14a}$$

$$\tilde{z}_t = C\tilde{x}_t, \tag{14b}$$

$$y_t = \tilde{h}(\tilde{z}_t) + e_t, \qquad e_t \sim \mathcal{N}(0, r), \tag{14c}$$

Algorithm 1 Wiener system identification using PG-BSi

Initialise: Set $A[0] = M$, $Q[0] = S_0$, $r[0] = R_0$ and $\mathbf{h}_\star[0] = \mathbf{z}_\star$. Set $X_{1:T}[0]$ and $J_{1:T}[0]$ arbitrarily.

For $i \geq 1$, iterate:

1. Sample, using (10), (11) and (12),
   (a) $\{A[i], Q[i]\} \sim p(A, Q \mid X_{1:T}[i-1])$.
   (b) $r[i] \sim p(r \mid \mathbf{h}_\star[i-1], X_{1:T}[i-1], y_{1:T})$.
   (c) $\mathbf{h}_\star[i] \sim p(\mathbf{h}_\star \mid r[i], X_{1:T}[i-1], y_{1:T})$
2. Set $\theta[i] = \{A[i], Q[i], r[i], \mathbf{h}_\star[i]\}$.
3. Run a CPF [Lindsten and Schön, 2012, Algorithm 2], targeting $p(x_{1:T} \mid \theta[i], y_{1:T})$ and conditioned on $\{X_{1:T}[i-1], J_{1:T}[i-1]\}$.
4. Run a backward simulator [Lindsten and Schön, 2012, Algorithm 1] to generate $J_{1:T}[i]$. Set $X_{1:T}[i]$ to the corresponding particle trajectory.

where we have defined $\tilde{Q} = c^2Q$ and $\tilde{h}(s) = h(s/c)$. Hence, there is a scale ambiguity in the model—we can "move" the constant c back and forth between the linear block and the static nonlinearity. When dealing with a single realisation of the model, this is typically not a problem. However, in the Gibbs sampler used in this work, we aim to approximate the posterior distribution of the unknowns of the model by a large number of random realisation of these quantities. It then becomes important that we, in some sense, use the same scale in all realisations.

Here, we address this problem by setting the range of the sequence $z_{1:T}$ at each iteration of the Gibbs sampler to some fixed value. More precisely, assume that we have obtained the model variables $\{A, Q, r, h(\cdot)\}$ as well as a trajectory $z_{1:T} = \{Cx_1, \dots, Cx_T\}$ at some iteration of the Gibbs sampler. Let λ be some arbitrary, positive constant. We then compute

$$c = \frac{\lambda}{\max(z_{1:T}) - \min(z_{1:T})} \tag{15}$$

and define $\tilde{x}_t$, $\tilde{z}_t$, $\tilde{Q}$ and $\tilde{h}$ as above. It follows that $\max(\tilde{z}_{1:T}) - \min(\tilde{z}_{1:T}) = \lambda$. Hence, by modifying the state of the Markov chain to include $\tilde{Q}$, $\tilde{h}$ and $\tilde{x}_{1:T}$, rather than Q, h and $x_{1:T}$, the range of the input to the nonlinear function will be the same for all iterations of the Gibbs sampler. We have found this heuristic to resolve the scale ambiguity to provide good results, but alternative approaches are of course possible.

5. NUMERICAL ILLUSTRATION

In this section we apply the proposed method, summarised in Algorithm 1, to identify two synthetic Wiener systems. In Algorithm 1 the variables $J_{1:T}$ refer to particles indices defining a state trajectory, generated by the backward simulator. See [Lindsten and Schön, 2012] for all details.

5.1 4th-order system with saturation

Consider a 4th-order linear dynamical system according to (1) with

$$A = \begin{pmatrix} 0.3676 & 0.88746 & 0.52406 & 0.55497 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \tag{16a}$$

$$C = \begin{pmatrix} 1 & 0.1 & -0.49 & 0.01 \end{pmatrix}, \tag{16b}$$

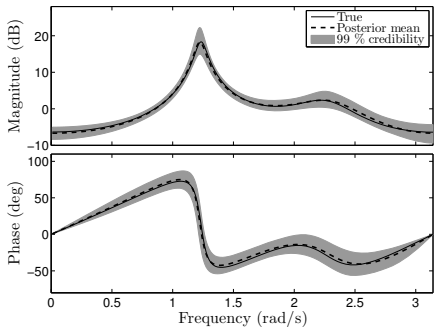

Fig. 3. Bode diagram of the linear system, estimated posterior mean and 99 % credibility interval.

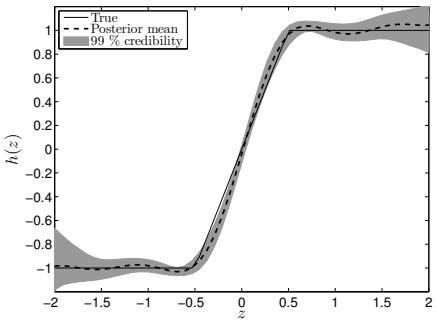

Fig. 4. Nonlinear mapping (saturation), estimated posterior mean and 99 % credibility interval.

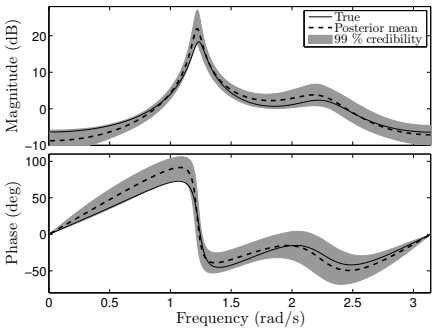

Fig. 5. Bode diagram of the linear system, estimated posterior mean and 99 % credibility interval.

$Q = 0.05I_4$ and $R = 0.01$. The same system is considered by Wills et al. [2011] who present a method for maximum likelihood estimation of blind Wiener systems. The nonlinear mapping $h(\cdot)$ is taken as a saturation,

$$h(z) = \begin{cases} 1 & \text{if } z \geq 0.5, \\ 2z & \text{if } -0.5 \leq z < 0.5, \\ -1 & \text{if } z < -0.5. \end{cases} \tag{17}$$

We generate $T = 1000$ samples from the system and apply the proposed method for 50000 MCMC iterations (out of which 10000 iterations are considered as burnin), using $N = 5$ particles in the PG-BSi sampler. The hyperparameters are set as described in Appendix B. Figure 3 shows the Bode diagram of the linear system and Figure 4 shows the static nonlinearity, along with their estimates. The gray areas illustrate the 99 % Bayesian credibility regions, computed from the posterior PDFs.

The method appears to do a good job at identifying both the linear dynamical system and the nonlinear mapping. Some slight lack of fit arises due to the non-smoothness of h and the fact that the GP is a smoothness prior. Still, the shape of the nonlinearity is clearly visible from the estimated posterior PDF. The uncertainty about the nonlinearity gets larger close to the border of the axis ($\|z\| \gtrsim 1.5$). The reason for this is that there are few samples in these regions available in the observed measurements.

5.2 4th-order system with non-monotone nonlinearity

To show the flexibility of the GP model, we consider the same linear system (16), but replace the static nonlinearity. Instead of the saturation, we use a non-monotonic nonlinear function shown in Figure 2. We generate $T = 1000$ observations from the system and apply the proposed identification method with the same settings as in Section 5.1. Note that, due to the nonparametric nature of the GP model, we do not need to make any modifications to the code when we apply it to this modified system. Figure 5 shows the Bode diagram of the linear system and Figure 2 shows the static nonlinearity, together with the estimates using 50000 MCMC iterations (out of which 10000 iterations are considered as burnin).

Also for this example, the method captures the shape of the nonlinearity as well as the linear dynamical system. The uncertainty about the Bode diagram is somewhat larger than in Section 5.1, which is reflected in the estimated posterior PDF. This is not surprising, since the nonlinearity illustrated in Figure 2 is quite difficult to deal with. The reason is that the non-monotonicity of the function means that there is an ambiguity of the value of z_t for a given observation y_t. Basically, for any observation y_t in the range $[-1, 1]$ there are three possible values for z_t which descibe this observation equally well statically.

6. CONCLUSIONS AND FUTURE WORK

We have considered a semiparametric Bayesian model of a Wiener system, using a state-space representation (where the dimension of the state-space is assumed to be known) of the linear dynamical system and a GP model for the static nonlinearity. The posterior parameter distribution is not available in closed form. This was resolved by making use of a particle Markov Chain Monte Carlo method, relying on a particle filter and a backward simualtor to produce sample state trajectories. The new algorithm was profiled on two examples with good results, despite the fact that only 5 particles were used in the underlying particle filter.

A concern with the current method is that it does not scale well with the number of measurements T, since the computational complexity of evaluating the posterior GP is cubic in T. However, this is a well-studied problem in the GP literature and existing approaches can be used to mitigate this issue.

In the numerical example provided in Section 5.1 we applied the method to estimate a Wiener model, where the true nonlinearity was given by a saturation. This system is in fact not contained in the proposed model class, since the GP that we use is a smoothness prior. Due to this, some problems arise close to the points of non-differentiability of the saturation. Still, the method captures the shape of this nonlinearity fairly well. An interesting topic for future work is to seek some theoretical justification for the application of the method, even when the true system lies outside the treated model class. We may also consider to use the proposed method in a prestudy of the identification problem. Once we find the rough shape of the nonlinearity, we can find some suitable parameterisation of it and switch to a fully parametric model.

In this work, we have not considered estimation of the GP hyperparameters from data. However, this can be done by adding a step to the Gibbs sampler, in which

the hyperparameters are sampled. We have found this to give good results (not reported here) and are underway of incorporating such a step into the proposed method. We are also currently in the process of developing a method where the dimension of the linear state-space is found directly from the data. Together with the results presented here, this will result in a fully automatic method, where the only structural assumption made is that we are looking for a Wiener model.

Appendix A. SEQUENTIAL GP SAMPLING

Assume that we wish to evaluate the function h at the points $\mathbf{z}_{\star,t}$ for each time $t = 1, \ldots, T$, i.e. according to the first alternative suggested in Section 4.3. Hence, we wish to sample according to $\mathbf{h}_{\star,t} \sim p(\mathbf{h}_{\star,t} \mid \mathbf{h}_{\star,1:t-1}, r, x_{1:T}, y_{1:T})$. We now describe how this can be done without resorting to operations of order $O(t^3)$ at each iteration.

Let (X, Y) be jointly Gaussian with density,

$$p(x, y) = \mathcal{N}\left(\begin{bmatrix} x \\ y \end{bmatrix}; \begin{bmatrix} m_x \\ m_y \end{bmatrix}, \begin{bmatrix} P_{xx} & P_{xy} \\ P_{xy}^\mathsf{T} & P_{yy} \end{bmatrix}\right) \tag{A.1}$$

We seek the conditional of Y given X. Let $n_x = \dim(X)$ and $n_y = \dim(Y)$. We assume that, in general, $n_x \gg n_y$. In the problem of sampling from the posterior GP at time t, X corresponds to the collection of variables $\{\mathbf{h}_{\star,1:t-1}, y_{1:T}\}$ and Y corresponds to $\mathbf{h}_{\star,t}$. Similarly to (12), the sought conditional density is given by $p(y \mid x) = \mathcal{N}(y; \mu, \Sigma)$ with,

$$\mu = m_y + P_{xy}^\mathsf{T} P_{xx}^{-1}(x - m_x), \tag{A.2a}$$

$$\Sigma = P_{yy} - P_{xy}^\mathsf{T} P_{xx}^{-1} P_{xy}. \tag{A.2b}$$

A straightforward evaluation of the mean and covariance above is of order n_x^3. Since n_x will increase with t, we seek a recursion in which the evaluation at time t is based on the ones from time $t-1$. Here we propose to use a recursion of the Cholesky factors of the covariance. Assume that we are *given* a Cholesky factorisation, $P_{xx} = R_x^\mathsf{T} R_x$. To compute the conditional mean (A.2a), let,

$$r_x \triangleq P_{xx}^{-1}(x - m_x) \Rightarrow R_x^\mathsf{T} R_x r_x = x - m_x. \tag{A.3}$$

Furthermore, define $s_x \triangleq R_x r_x$. We can then compute r_x by solving the linear systems of equations $R_x^\mathsf{T} s_x = x - m_x$ and $R_x r_x = s_x$. Since R_x is triangular, this can be done in $O(n_x^2)$ by using back-substitution. It follows that the conditional mean (A.2a) is given by $\mu = m_y + P_{xy}^\mathsf{T} r_x$.

To compute the conditional covariance, consider a Cholesky factorisation of the joint covariance matrix of X and Y,

$$\begin{bmatrix} P_{xx} & P_{xy} \\ P_{xy}^\mathsf{T} & P_{yy} \end{bmatrix} = R_{xy}^\mathsf{T} R_{xy} = \begin{bmatrix} \chi_{11}^\mathsf{T} & 0 \\ \chi_{12}^\mathsf{T} & \chi_{22}^\mathsf{T} \end{bmatrix} \begin{bmatrix} \chi_{11} & \chi_{12} \\ 0 & \chi_{22} \end{bmatrix}. \tag{A.4}$$

If follows that $\chi_{11} = R_x$. We can obtain χ_{12} by solving the system of equations $R_x^\mathsf{T} \chi_{12} = P_{xy}$, which can be done by back-substitution in $O(n_x^2 n_y)$. Finally, χ_{22} is given by a Cholesky factorisation of $\chi_{22}^\mathsf{T} \chi_{22} = P_{yy} - \chi_{12}^\mathsf{T} \chi_{12}$, which can be done in $O(n_y^3)$ for the factorisation and $O(n_x n_y^2)$ for computing the right hand side.

To obtain the conditional covariance (A.2b), we note that

$$\begin{aligned} \chi_{22}^\mathsf{T} \chi_{22} &= P_{yy} - \chi_{12}^\mathsf{T} \chi_{12} = P_{yy} - \chi_{12}^\mathsf{T} \chi_{11} \chi_{11}^{-1} (\chi_{11}^{-1})^\mathsf{T} \chi_{11}^\mathsf{T} \chi_{12} \\ &= P_{yy} - P_{xy}^\mathsf{T} (\chi_{11}^\mathsf{T} \chi_{11})^{-1} P_{xy} = \Sigma. \end{aligned} \tag{A.5}$$

Hence, the conditional covariance is given directly from the Cholesky factorisation. In summary, the cost of computing the conditional mean and covariance, as well as updating the Cholesky factor, is of order $O(n_x^2 n_y + n_x n_y^2 + n_y^3)$.

Appendix B. CHOOSING THE HYPERPARAMETERS

To tune the hyperparameters we use an approach known as empirical Bayes, in which we use the observations $y_{1:T}$ to set the priors. We use the following heuristic. First, we run a subspace identification algorithm on the data (see e.g. Van Overschee and De Moor [1996]). The resulting state-space model is transformed into observer canonical form. We then set the mean M of the MN prior (2a) to the resulting A-matrix. The covariance L^{-1} is set to identity. This choice allows for a considerable variability around the mean. For the IW priors (2b) and (3) we use the same heuristic as [Fox, 2009, p. 156–160], based on the empirical covariance of the observations $y_{1:T}$. Finally, for the GP prior we have used unit hyperparameters $\alpha = \ell = 1$ for the covariance kernel (5). Note that the length-scale of the GP kernel is used to control the scale of the function h. However, as discussed in Section 4.4, the scale is fixed by the user. Hence, we can set the scale by choosing λ, and then choose the length-scale of the GP kernel to correspond to the expected variability of h.

REFERENCES

K. Abed-Meraim, W. Qiu, and Y. Hua. Blind system identification. *Proceedings of the IEEE*, 85(8):1310–1322, 1997.

C. Andrieu, A. Doucet, and R. Holenstein. Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B*, 72(3):269–342, 2010.

E-W Bai. A blind approach to Hammerstein-Wiener model identification. *Automatica*, 38(6):967–979, 2002.

R. Douc, A. Garivier, E. Moulines, and J. Olsson. Sequential Monte Carlo smoothing for general state space hidden Markov models. *Annals of Applied Probability*, 21(6):2109–2145, 2011.

Emily B. Fox. *Bayesian Nonparametric Learning of Complex Dynamical Phenomena*. PhD thesis, Massachusetts Institute of Technology, 2009.

F. Giri and E-W. Bai, editors. *Block-oriented Nonlinear System Identification*, volume 404 of *Lecture notes in control and information sciences*. Springer, 2010.

S. J. Godsill, A. Doucet, and M. West. Monte Carlo smoothing for nonlinear time series. *Journal of the American Statistical Association*, 99(465):156–168, March 2004.

A. Hammerstein. Nichtlineare integralgleichungen nebst anwendungen. *Acta Mathematica*, 54(1):117–176, 1930.

F. Lindsten and T. B. Schön. On the use of backward simulation in the particle Gibbs sampler. In *Proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Kyoto, Japan, March 2012.

G. Pillonetto and A. Chiuso. Gaussian processes for Wiener-Hammerstein system identification. In *Proceedings of the 15th IFAC Symposium on System Identification*, Saint-Malo, France, July 2009.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer, 2004.

P. Van Overschee and B. De Moor. *Subspace Identification for Linear Systems: Theory, Implementation, Applications*. Kluwer Academic Publishers, 1996.

L. R. Vanbeylan, R. Pintelon, and J. Schoukens. Blind maximum likelihood identification of Wiener systems. *IEEE Transactions on Signal Processing*, 57(8):3017–3029, 2009.

M. West and J. Harrison. *Bayesian Forecasting and Dynamic Models*. Springer, New York, 1997.

N. Whiteley. Discussion on Particle Markov chain Monte Carlo methods. Journal of the Royal Statistical Society: Series B, 72(3), p 306–307, 2010.

N. Wiener. *Nonlinear Problems in Random Theory*. The MIT Press, Cambridge, MA, USA, 1966.

A. Wills, T. B. Schön, L. Ljung, and B. Ninness. Blind identification of Wiener models. In *Proceedings of the 18th IFAC World Congress*, Milan, Italy, August 2011.