

# Identification of Mixed Linear/Nonlinear State-Space Models

Fredrik Lindsten and Thomas B. Schön

**Abstract**—The primary contribution of this paper is an algorithm capable of identifying parameters in certain mixed linear/nonlinear state-space models, containing conditionally linear Gaussian substructures. More specifically, we employ the standard maximum likelihood framework and derive an expectation maximization type algorithm. This involves a nonlinear smoothing problem for the state variables, which for the conditionally linear Gaussian system can be efficiently solved using a so called Rao-Blackwellized particle smoother (RBPS). As a secondary contribution of this paper we extend an existing RBPS to be able to handle the fully interconnected model under study.

## I. INTRODUCTION

Identification of nonlinear systems is probably one of the currently most active areas within the system identification community [1, 2]. This is basically due to its relevance and challenging nature. During the last decade or so, identification methods based on the so called Sequential Monte Carlo (SMC) method, also referred to as particle filters [3, 4], have appeared at an increasing rate and with increasingly better performance. The two overview papers [5, 6] and the recent results in [7–10] provide a good introduction to these ideas. We will in this paper continue this line of work and introduce a new algorithm based on SMC methods for nonlinear system identification.

More specifically, the main contribution of this paper is an algorithm for identifying mixed linear/nonlinear state-space models in the following form

$$a_{t+1} = f_a(a_t, u_t, \theta) + A_a(a_t, u_t, \theta)z_t + w_{a,t}, \quad (1a)$$

$$z_{t+1} = f_z(a_t, u_t, \theta) + A_z(a_t, u_t, \theta)z_t + w_{z,t}, \quad (1b)$$

$$y_t = h(a_t, u_t, \theta) + C(a_t, u_t, \theta)z_t + e_t, \quad (1c)$$

where  $x_t = (a_t^T \ z_t^T)^T \in \mathbb{R}^{n_x}$ ,  $u_t \in \mathbb{R}^{n_u}$ ,  $y_t \in \mathbb{R}^{n_y}$  denote the state, the measured input signal and the measured output signal, respectively. The model is parameterized by  $\theta \in \mathbb{R}^{n_\theta}$ . The random processes  $w_t = (w_{a,t}^T \ w_{z,t}^T)^T$  and  $e_t$  are assumed to be white and Gaussian according to

$$w_t \sim \mathcal{N}(0, Q(a_t, \theta)), \quad e_t \sim \mathcal{N}(0, R(a_t, \theta)), \quad (1d)$$

$$Q(a_t, \theta) = \begin{bmatrix} Q_a(a_t, \theta) & Q_{az}(a_t, \theta) \\ (Q_{az}(a_t, \theta))^T & Q_z(a_t, \theta) \end{bmatrix}. \quad (1e)$$

Furthermore, the initial state  $z_1$  is assumed Gaussian according to  $z_1 \sim \mathcal{N}(z_{1|0}(a_1, \theta), P_{1|0}(a_1, \theta))$  and the density of  $a_1$ ,  $p(a_1, \theta)$ , is assumed to be of known structure, but can

depend on  $\theta$ . We could also straightforwardly allow time-varying models, but to keep the notation simple we shall not make this dependence explicit.

Maximum Likelihood (ML) estimation has, due to its appealing theoretical properties, a long tradition in many fields of science, including the field of system identification. The ML problem we are concerned with in this work is

$$\hat{\theta} = \arg \max_{\theta \in \Theta} p_\theta(Y_N) = \arg \max_{\theta \in \Theta} L_\theta(Y_N), \quad (2)$$

where  $Y_N \triangleq \{y_1, y_2, \dots, y_N\}$ ,  $L_\theta(Y_N) \triangleq \log p_\theta(Y_N)$  is the log-likelihood function and  $\Theta \subset \mathbb{R}^{n_\theta}$  denotes a set of permissible values for the unknown parameter vector  $\theta$ . For future reference we also introduce the notation  $y_{s:t} \triangleq \{y_s, y_{s+1}, \dots, y_t\}$ .

The EM algorithm [11] is by now a standard tool for solving maximum likelihood problems. Despite this, it is only recently that it has gained serious interest in solving system identification problems [7, 9, 12].

When it comes to nonlinear problems, an appealing property of the EM algorithm is that it allows for straightforward application of SMC methods. This is thoroughly explained in [7], but basically it comes down to the fact that we have to compute conditional expectations of the form

$$\mathbb{E}\{g(x_{t:t+1}) \mid Y_N\} = \int g(x_{t:t+1})p(x_{t:t+1} \mid Y_N)dx_{t:t+1}, \quad (3)$$

for arbitrary functions  $g(\cdot)$ . Here, it is the presence of the smoothing density function  $p(x_{t:t+1} \mid Y_N)$  that opens up for direct use of SMC methods. These quantities could of course be computed using standard particle smoothers, similar to what was done in [7], not acknowledging the conditionally linear Gaussian substructure available in (1). However, as we will show in this paper, by exploiting this inherent structure, better results can be obtained.

A secondary contribution of this paper is a so called Rao-Blackwellized particle smoother (RBPS) capable of estimating the smoothing density  $p(a_{t:t+1}, z_{t:t+1} \mid Y_N)$ , required in (3), by exploiting the conditionally linear Gaussian substructure inherent in (1). Several RBPS algorithms have already been published [13, 14]. However, to the best of the authors' knowledge, this is the first time an algorithm capable of handling the more general model structure (1) is presented.

## II. RELATED WORK

This section provides an overview of the existing work on using SMC methods, which during the last decade have experienced a rapid increase in popularity, for solving the nonlinear system identification problem. A recent and more thorough overview is provided in [6].

F. Lindsten and T. B. Schön are with the Division of Automatic Control, Linköping University, SE-581 83 Linköping, Sweden, E-mail: (lindsten, schon)@isy.liu.se

This work was supported by the strategic research center MOVIII, funded by the Swedish Foundation for Strategic Research (SSF) and CADICS, a Linneaus Center funded by the Swedish Research Council.

When it comes to solving the ML problem (2) using SMC methods the literature can roughly be divided into two different categories,

- Direct maximization of the log-likelihood function. There are some methods only working with the function values, see e.g. [15]. However, most methods exploit a quadratic or linear model of the cost function  $L_\theta(Y)$ , rendering the computation of gradients and Hessians necessary [10, 16]. This is a challenging problem, since these objects are not easily approximated. However, interesting developments, based on SMC, are available in [5, 10, 16].
- Methods based on the EM algorithm [5, 7–9]. In this approach we refrain from working directly with the log-likelihood. Instead, as already mentioned in the introduction, we need to compute certain conditional expectations which straightforwardly can be approximated using standard SMC methods. This is explained in detail in for example [7] and it will also be elaborated in the subsequent development of the present work.

There are also some interesting developments based on the Bayesian approach. The most commonly used Bayesian approach is probably the idea of treating the parameter as a random walk state variable [17],  $\theta_{t+1} = \theta_t + v_t$ ,  $v_t \sim \mathcal{N}(0, V_t)$ , where the noise covariance  $V_t$  is decreasing with time  $t$ . This variable is then augmented to the state  $x_t$ , forming an augmented model. We [18] have previously used this idea for estimating parameters in models of the form (1).

### III. MAXIMUM LIKELIHOOD USING EM

The EM algorithm [11] is by now a standard tool in solving maximum likelihood problems. There are, as we have seen in Section II, by now quite a few publications containing general derivations of the EM algorithm with the problem of system identification in mind, see e.g. [7, 9, 12, 19]. Hence, we refrain from repeating that here and simply point the reader to the above references for such a derivation.

However, it is still necessary to define some basic quantities for mixed linear/nonlinear systems on the form (1). Let us start by defining the latent variables (sometimes also referred to as the missing data or the hidden variables) according to  $X_N \triangleq \{x_1, x_2, \dots, x_N\}$ . The EM algorithm is an iterative method that finds the ML solution by alternatively computing (an approximation of) the  $\mathcal{Q}$ -function,

$$\mathcal{Q}(\theta, \theta_k) \triangleq \mathbb{E}_{\theta_k} \{L_\theta(X_N, Y_N) \mid Y_N\} \quad (4a)$$

$$= \int L_\theta(X_N, Y_N) p_{\theta_k}(X_N \mid Y_N) dX_N, \quad (4b)$$

and solving the following maximization problem

$$\theta_{k+1} = \arg \max_{\theta \in \Theta} \mathcal{Q}(\theta, \theta_k). \quad (5)$$

We will throughout the rest of this section assume that approximations of  $p(a_{t:t+1}, z_{t:t+1} \mid Y_N)$  are available in the form

$$\begin{aligned} \hat{p}(a_{t:t+1}, z_{t:t+1} \mid Y_N) &= \frac{1}{M} \sum_{j=1}^M \delta(a_{t:t+1} - \tilde{a}_{t:t+1}^j) \times \\ &\mathcal{N}(z_{t:t+1}; \tilde{z}_{t:t+1|N}^j, \tilde{P}_{t:t+1|N}^j). \quad (6) \end{aligned}$$

The notation used in (6) will be introduced in Section IV, where an algorithm for computing these approximations will be provided. Based on (6) we can compute an approximation of the  $\mathcal{Q}$ -function, which is the topic of Section III-A. The resulting EM algorithm is then provided in Section III-B.

#### A. Approximating the $\mathcal{Q}$ -Function

According to the definition of the  $\mathcal{Q}$ -function (4a) it depends on the joint log-likelihood function  $L_\theta(X_N, Y_N)$ , which for the model (1) under consideration is given by

$$\begin{aligned} L_\theta(X_N, Y_N) &= \log p_\theta(Y_N \mid X_N) + \log p_\theta(X_N) \\ &= \log p_\theta(x_1) + \sum_{t=1}^{N-1} \log p_\theta(x_{t+1} \mid x_t) + \sum_{t=1}^N \log p_\theta(y_t \mid x_t). \end{aligned} \quad (7)$$

From (4b) it appears as if we need the joint smoothing density for all the states,  $p(X_N \mid Y_N)$ . However, since the log-likelihood can be written according to (7) it is sufficient with marginal smoothing densities  $p(x_{t:t+1} \mid Y_N)$  in order to compute the  $\mathcal{Q}$ -function.

Inserting (7) into the definition of the  $\mathcal{Q}$ -function straightforwardly results in the following expression

$$\mathcal{Q}(\theta, \theta_k) = I_1(\theta, \theta_k) + I_2(\theta, \theta_k) + I_3(\theta, \theta_k), \quad (8)$$

where we have defined

$$I_1(\theta, \theta_k) \triangleq \mathbb{E}_{\theta_k} \{\log p_\theta(x_1) \mid Y_N\}, \quad (9a)$$

$$I_2(\theta, \theta_k) \triangleq \sum_{t=1}^{N-1} \mathbb{E}_{\theta_k} \{\log p_\theta(x_{t+1} \mid x_t) \mid Y_N\}, \quad (9b)$$

$$I_3(\theta, \theta_k) \triangleq \sum_{t=1}^N \mathbb{E}_{\theta_k} \{\log p_\theta(y_t \mid x_t) \mid Y_N\}. \quad (9c)$$

The fact that we in (1) have additive Gaussian noise allows us to derive explicit expressions for calculating the  $\mathcal{Q}$ -function. This is the topic throughout the rest of this section. In the interest of a clearer notation, let us start by rewriting (1) according to

$$x_{t+1} = f(a_t, \theta) + A(a_t, \theta)z_t + w_t, \quad (10a)$$

$$y_t = h(a_t, \theta) + C(a_t, \theta)z_t + e_t, \quad (10b)$$

where

$$f(a_t, \theta) \triangleq \begin{bmatrix} f_a(a_t, \theta) \\ f_z(a_t, \theta) \end{bmatrix}, \quad A(a_t, \theta) \triangleq \begin{bmatrix} A_a(a_t, \theta) \\ A_z(a_t, \theta) \end{bmatrix}. \quad (11)$$

Here we have neglected the possible dependence of a known input  $u_t$  to keep the notation simple.

Let us start by considering the term  $I_2$  defined in (9b). Using the fact that  $x^T A x = \text{Tr}(A x x^T)$ , provides the following expression for the density function  $p_\theta(x_{t+1} \mid x_t)$ ,

$$\begin{aligned} -2 \log p_\theta(x_{t+1} \mid x_t) &= -2 \log p_{w, \theta}(x_{t+1} - f(a_t) - A(a_t)z_t) \\ &= \log \det Q(a_t) + \text{Tr}(Q^{-1}(a_t) \ell_2(a_{t:t+1}, z_{t:t+1})) + \text{const.}, \end{aligned} \quad (12)$$

where

$$\begin{aligned} \ell_2(a_{t:t+1}, z_{t:t+1}) &\triangleq (x_{t+1} - f(a_t) - A(a_t)z_t) \times \\ &(x_{t+1} - f(a_t) - A(a_t)z_t)^T. \end{aligned} \quad (13)$$

For notational convenience we have dropped the dependence on  $\theta$ , but remember that all of  $f$ ,  $A$ ,  $Q$ ,  $h$ ,  $C$ ,  $R$ ,  $z_{1|0}$ ,  $P_{1|0}$  and  $p(a_1)$  may in fact be  $\theta$ -dependent.

Inserting this into (9b) results in

$$I_2 = -\frac{1}{2} \sum_{t=1}^{N-1} \left( \mathbb{E}_{\theta_k} \left\{ \text{Tr} \left( Q^{-1}(a_t) \ell_2(a_{t:t+1}, z_{t:t+1}) \right) \mid Y_N \right\} + \mathbb{E}_{\theta_k} \left\{ \log \det Q(a_t) \mid Y_N \right\} \right) + \text{const.} \quad (14)$$

Except for simple special cases, such as linear Gaussian models, it is impossible to obtain explicit solutions for the involved conditional expectations. However, using (6) we can compute arbitrarily good approximations according to

$$\hat{I}_2 = -\frac{1}{2M} \sum_{t=1}^{N-1} \sum_{j=1}^M \left( \log \det Q(\tilde{a}_t^j) + \text{Tr} \left( Q^{-1}(\tilde{a}_t^j) \hat{\ell}_{2,t}^j \right) \right), \quad (15)$$

where

$$\hat{\ell}_{2,t}^j \triangleq \mathbb{E}_{\theta_k} \left\{ \ell_2(\tilde{a}_{t:t+1}^j, z_{t:t+1}) \mid \tilde{a}_{t:t+1}^j, Y_N \right\}. \quad (16)$$

Observe that the expectation here only is taken over the  $z$ -variables, conditioned on the  $a$ -variables. The nontrivial parts of the above conditional expectation are the terms

$$\mathbb{E}_{\theta_k} \left\{ z_t z_t^T \mid \tilde{a}_{t:t+1}^j, Y_N \right\} = \tilde{z}_{t|N}^j \tilde{z}_{t|N}^{jT} + \tilde{P}_{t|N}^j, \quad (17a)$$

$$\mathbb{E}_{\theta_k} \left\{ z_t z_{t+1}^T \mid \tilde{a}_{t:t+1}^j, Y_N \right\} = \tilde{z}_{t|N}^j \tilde{z}_{t+1|N}^{jT} + M_{t|N}^j, \quad (17b)$$

where  $\tilde{z}_{t|N}^j$ ,  $\tilde{P}_{t|N}^j$  and  $M_{t|N}^j$  are provided in Section IV. The rest of the computations involved in finding (16) are straightforward and in the interest of saving space, we refrain from providing all the details here. Analogously, for  $I_1$  and  $I_3$ , we obtain

$$\hat{I}_1 = -\frac{1}{2M} \sum_{j=1}^M \left( \log \det P_{1|0}(\tilde{a}_1^j) + \text{Tr} \left( P_{1|0}^{-1}(\tilde{a}_1^j) \hat{\ell}_1^j \right) \right) + \frac{1}{M} \sum_{j=1}^M \log p(\tilde{a}_1^j), \quad (18a)$$

$$\hat{I}_3 = -\frac{1}{2M} \sum_{t=1}^N \sum_{j=1}^M \left( \log \det R(\tilde{a}_t^j) + \text{Tr} \left( R^{-1}(\tilde{a}_t^j) \hat{\ell}_{3,t}^j \right) \right) \quad (18b)$$

where

$$\hat{\ell}_1^j \triangleq \mathbb{E}_{\theta_k} \left\{ (z_1 - z_{1|0}(\tilde{a}_1^j))(z_1 - z_{1|0}(\tilde{a}_1^j))^T \mid \tilde{a}_1^j, Y_N \right\}, \quad (19a)$$

$$\hat{\ell}_{3,t}^j \triangleq \mathbb{E}_{\theta_k} \left\{ (y_t - h(\tilde{a}_t^j) - C(\tilde{a}_t^j)z_t) \times (y_t - h(\tilde{a}_t^j) - C(\tilde{a}_t^j)z_t)^T \mid \tilde{a}_t^j, Y_N \right\}. \quad (19b)$$

## B. Final Algorithm

We summarize the above calculations in Algorithm 1.

*Algorithm 1 (RBPS-EM):*

- 1) **Initialize:** Let  $k = 1$  and initialize the parameter estimate  $\theta_1$ .
- 2) **Expectation (E) step:**
  - a) *Smoothing:* Parameterize the model (1) using the current parameter estimate,  $\theta_k$ . Run the

RBPS (See Algorithm 2) and store the particles and the sufficient statistics for the linear states  $\{\tilde{a}_{t:N}^j, \tilde{z}_{t|N}^j, \tilde{P}_{t|N}^j\}_{j=1}^M$  for  $t = 1, \dots, N$  and  $\{M_{t|N}^j\}_{j=1}^M$  for  $t = 1, \dots, N-1$ .

- b) *Approximate the Q-function:* Let

$$\hat{Q}(\theta, \theta_k) = \hat{I}_1(\theta, \theta_k) + \hat{I}_2(\theta, \theta_k) + \hat{I}_3(\theta, \theta_k),$$

where  $\hat{I}_1$ ,  $\hat{I}_2$  and  $\hat{I}_3$  are given by (18a), (15) and (18b), respectively.

- 3) **Maximization (M) step:** Compute

$$\theta_{k+1} = \arg \max_{\theta \in \Theta} \hat{Q}(\theta, \theta_k)$$

- 4) **Termination condition:** If not converged, set  $k \leftarrow k + 1$  and go to step 2, otherwise terminate.

The M step can be carried out using any appropriate optimization routine. For all examples presented in this paper a BFGS Quasi-Newton method was used (see e.g. [20]).

## IV. A RAO-BLACKWELLIZED PARTICLE SMOOTHER

As previously pointed out, a crucial step of the proposed identification algorithm is to compute conditional expectations of the form (3). This can, for general nonlinear state-space models, be done using so called particle smoothers, see e.g. [4, 14]. However, when there is a conditionally linear Gaussian substructure present in the model, as is the case in (1), it is of interest to exploit this to reduce the variance of the Monte Carlo approximation. This can be done by marginalizing the conditionally linear Gaussian states (i.e. the part of the state vector here denoted by  $z_t$ ). This approach is often called Rao-Blackwellization after the Rao-Blackwell theorem (see [21]) and we shall hence call a smoother that exploits it a Rao-Blackwellized Particle Smoother (RBPS).

Several RBPS have been presented in the literature [13, 14]. However, both of these assume a different model structure and can only be used for model (1) in the special case  $A_a \equiv Q_{az} \equiv 0$ , i.e. when the nonlinear state dynamics are independent of the linear states. In this section we will provide an extension of the smoother derived in [13], capable of handling the fully interconnected model (1). The derivation that follows is rather brief and for all the details we refer to [22].

The idea is to run a Rao-Blackwellized Particle Filter (RBPF) forward in time (see e.g. [4, 23]) and then carry out the smoothing by simulating trajectories backward in time. We shall assume that we already have performed the forward filtering (FF) and, for  $t = 1, \dots, N$ , have approximations of the filtering distribution according to

$$\hat{p}(z_t, a_t \mid Y_t) = \sum_{i=1}^M w_t^i \mathcal{N} \left( z_t; z_{t|t}^i, P_{t|t}^i \right) \delta(a_t - a_t^i). \quad (20)$$

The task at hand is now to, starting at time  $t = N$ , backward in time sample trajectories  $\tilde{a}_{t:N}^j$  in the nonlinear state dimension and for each trajectory evaluate the sufficient statistics for the linear states. At time  $t = N$  we can do this simply by resampling the FF. Thus, assume that we have completed the smoothing recursion at time  $t+1$ , i.e. we have the trajectories

and the sufficient statistics  $\{\tilde{a}_{t+1:N}^j, \tilde{z}_{t+1|N}^j, \tilde{P}_{t+1|N}^j\}_{j=1}^M$ . For each trajectory we wish to append a sample

$$\tilde{a}_t^j \sim p(a_t | \tilde{a}_{t+1:N}^j, Y_N). \quad (21)$$

It turns out that it is in fact easier to sample from the joint distribution (see [22])

$$\begin{aligned} p(z_{t+1}, a_t | \tilde{a}_{t+1:N}^j, Y_N) \\ = p(a_t | z_{t+1}, \tilde{a}_{t+1:N}^j, Y_N) \mathcal{N}(z_{t+1}; \tilde{z}_{t+1|N}^j, \tilde{P}_{t+1|N}^j) \end{aligned} \quad (22)$$

by first drawing  $\tilde{Z}_{t+1}^j$  from the second factor in (22) and thereafter sample from

$$p(a_t | \tilde{Z}_{t+1}^j, \tilde{a}_{t+1:N}^j, Y_N) \propto p(\tilde{Z}_{t+1}^j, \tilde{a}_{t+1}^j | a_t, Y_t) p(a_t | Y_t). \quad (23)$$

This density is, using (20), approximately given by

$$p(a_t | \tilde{Z}_{t+1}^j, \tilde{a}_{t+1:N}^j, Y_N) \approx \sum_{i=1}^M w_{t|N}^{i,j} \delta(a_t - a_t^i), \quad (24)$$

where

$$w_{t|N}^{i,j} \propto w_t^i p(\tilde{Z}_{t+1}^j, \tilde{a}_{t+1}^j | a_t^i, Y_t), \quad \sum_{i=1}^M w_{t|N}^{i,j} = 1. \quad (25)$$

From the FF we can obtain the density involved in the above expression (using the model description (10a)) as

$$\begin{aligned} p(z_{t+1}, a_{t+1} | a_t^i, Y_t) \\ = \mathcal{N}(x_{t+1}; f^i + A^i z_{t|t}^i, Q^i + A^i P_{t|t}^i (A^i)^T). \end{aligned} \quad (26)$$

Here we have employed the shorthand notation,  $f^i = f(a_t^i)$  etc. Hence, we can compute the weights (25) and sample  $\tilde{a}_t^j$  from (24) as  $P(\tilde{a}_t^j = a_t^i) = w_{t|N}^{i,j}$ .

Once we have appended the new sample to the trajectory,  $\tilde{a}_{t:N}^j := \{\tilde{a}_t^j, \tilde{a}_{t+1:N}^j\}$ , we must update the expressions for the sufficient statistics of the linear states. Since we also need to compute expectations of the form (17b), we also require an expression for

$$M_{t|N}^j \triangleq \text{Cov}\{z_t z_{t+1}^T | \tilde{a}_{t:N}^j, Y_N\}. \quad (27)$$

As a stepping stone, let us start by considering

$$p(z_t | z_{t+1}, a_{t:N}, Y_N) = p(z_t | z_{t+1}, a_t, a_{t+1}, Y_t), \quad (28)$$

where the equality follows from the Markov property of the model. The right hand side in (28) can be recognized as the filtering distribution for the linear states, but conditioned on the ‘‘future’’ state  $x_{t+1}$ . From the filtering distribution,  $p(z_t | a_t^i, Y_t) = \mathcal{N}(z_t; z_{t|t}^i, P_{t|t}^i)$  and the dynamics of the system (10a) we can find an expression for (28) which turns out to be Gaussian and its mean is affine in  $z_{t+1}$ ,

$$p(z_t | z_{t+1}, a_t^i, Y_t) = \mathcal{N}(z_t; \Sigma_{t|t}^i W_z^i z_{t+1} + c_{t|t}^i(a_{t+1}), \Sigma_{t|t}^i) \quad (29)$$

where we have defined

$$[W_a(a_t) \quad W_z(a_t)] \triangleq A(a_t)^T Q^{-1}(a_t), \quad (30a)$$

$$\Sigma_{t|t}^i \triangleq P_{t|t}^i - P_{t|t}^i (A^i)^T \left( Q^i + A^i P_{t|t}^i (A^i)^T \right)^{-1} A^i P_{t|t}^i, \quad (30b)$$

$$c_{t|t}^i \triangleq \Sigma_{t|t}^i \left( W_a^i(a_{t+1} - f_a^i) - W_z^i f_z^i + (P_{t|t}^i)^{-1} z_{t|t}^i \right). \quad (30c)$$

Now, from (28) and (29) we see that the density

$$p(z_t | z_{t+1}, a_t^i, \tilde{a}_{t+1:N}^j, Y_N) \quad (31)$$

is Gaussian and affine in  $z_{t+1}$ . Furthermore, let us approximate

$$\begin{aligned} p(z_{t+1} | a_t^i, \tilde{a}_{t+1:N}^j, Y_N) &\approx p(z_{t+1} | \tilde{a}_{t+1:N}^j, Y_N) \\ &= \mathcal{N}(z_{t+1}; \tilde{z}_{t+1|N}^j, \tilde{P}_{t+1|N}^j). \end{aligned} \quad (32)$$

The approximation involved in (32) is discussed in detail in [22]. Basically it implies that we assume that the smoothing estimate for  $z_{t+1}$  is independent of which FF particle  $a_t^i$  that is appended to the backward trajectory.

Finally, we observe that the product of the densities (31) and (32) will yield the joint density for  $z_t$  and  $z_{t+1}$  conditioned on the backward trajectory  $\tilde{a}_{t:N}^j$  (where we have sampled  $\tilde{a}_t^j = a_t^i$ ) and the measurements  $Y_N$ . Since both densities are Gaussian and (31) is affine in  $z_{t+1}$ , their product will be Gaussian as well. Hence, from the joint distribution we can extract the sufficient statistics for  $z_t$  and the sought cross covariance (27), yielding

$$\tilde{z}_{t|N}^j = \Sigma_{t|t}^i W_z^i \tilde{z}_{t+1|N}^j + c_{t|t}^i(\tilde{a}_{t+1}^j), \quad (33a)$$

$$\tilde{P}_{t|N}^j = \Sigma_{t|t}^i + M_{t|N}^j (W_z^i)^T \Sigma_{t|t}^i, \quad (33b)$$

$$M_{t|N}^j = \Sigma_{t|t}^i W_z^i \tilde{P}_{t+1|N}^j. \quad (33c)$$

We summarize the resulting RBPS in Algorithm 2.

---

*Algorithm 2 (Rao-Blackwellized Particle Smoother):*

1) **Initialize:** Run a forward pass of the RBPF [23] and store the particles, the weights and the sufficient statistics for the linear states  $\{a_t^i, w_t^i, z_{t|t}^i, P_{t|t}^i\}_{i=1}^M$ . Resample the FF at time  $t = N$ , i.e.  $\{\tilde{a}_N^j, \tilde{z}_{N|N}^j, \tilde{P}_{N|N}^j\}_{j=1}^M := \{a_N^i, z_{N|N}^i, P_{N|N}^i\}$  with probability  $w_N^i$ .

2) **Backward simulation:** For  $t = N - 1$  to 1, do:

For  $j = 1, \dots, M$ ,

a) Sample  $\tilde{Z}_{t+1}^j \sim \mathcal{N}(z_{t+1}; \tilde{z}_{t+1|N}^j, \tilde{P}_{t+1|N}^j)$ .

b) Evaluate  $\{w_{t|N}^{i,j}\}_{i=1}^M$  from (25) and (26) and sample  $P(\tilde{a}_t^j = a_t^i) = w_{t|N}^{i,j}$ .

c) Update the sufficient statistics according to (33), where index  $i$  refers to the FF particle that was drawn in step 2b).

---

In a straightforward implementation, the above smoother has complexity  $\mathcal{O}(M^2)$ , which is typical for many particle smoothers. However, recent developments show that the complexity can be reduced to  $\mathcal{O}(M)$  [24, 25]. One interesting topic for future work would be to incorporate these ideas into the proposed estimation method.

## V. NUMERICAL ILLUSTRATIONS

In this section we will evaluate the proposed method on simulated data. Two different examples will be presented, first a linear Gaussian system and thereafter a nonlinear system. The purpose of including a linear Gaussian example is to gain confidence in the proposed method. This is possible, since, for this case, there are closed form solutions available for all the involved calculations (see [12] for all the details in a very similar setting). The smoothing densities for this case be explicitly calculated using the Rauch-Tung-Striebel (RTS) recursions [26]. We will refer to the resulting estimation algorithm as RTS-EM.

For both the linear and nonlinear examples, we can clearly also address the estimation problem using standard particle smoothing (PS) based methods, as in [7]. This approach will be denoted PS-EM. Finally, we have the option to employ the proposed RBPS-EM presented in Algorithm 1.

### A. A Linear Example

To test the proposed algorithm, we shall start by considering a linear, second order system with a single unknown parameter  $\theta$  according to

$$\begin{pmatrix} a_{t+1} \\ z_{t+1} \end{pmatrix} = \begin{pmatrix} 1 & \theta \\ 0 & 1 \end{pmatrix} \begin{pmatrix} a_t \\ z_t \end{pmatrix} + w_t, \quad Q = \begin{pmatrix} 0.1 & 0 \\ 0 & 0.1 \end{pmatrix} \quad (34a)$$

$$y_t = a_t + e_t, \quad R = 0.1 \quad (34b)$$

with the initial state  $(a_1 \ z_1)^T \sim \mathcal{N}\left(\begin{pmatrix} 0 & 1 \end{pmatrix}^T, I_{2 \times 2}\right)$ .

The comparison was made by pursuing a Monte Carlo study over 130 realizations of data  $Y_N$  from the system (34), each consisting of  $N = 200$  samples (measurements). The true value of the parameter was set to  $\theta^* = 0.1$ . The three estimation methods, RTS-EM, PS-EM and RBPS-EM were run in parallel for 100 iterations of the EM algorithm. The latter two both employed  $M = 50$  particles and the initial parameter estimate  $\theta_1$  was set to 0.3 in all experiments.

Table I gives the Monte Carlo means and standard deviations for the parameter estimates. On average, all methods converge rather close to the true parameter value 0.1. This is expected, since they all use ML which gives an unbiased estimate. The major difference is in the standard deviations of the estimated parameter. For RTS-EM and RBPS-EM, the standard deviations are basically identical, whereas for PS-EM it is more than twice as high.

TABLE I  
MONTE CARLO MEANS AND STANDARD DEVIATIONS

Method	Mean ( $\times 10^{-2}$ )	Std. dev. ( $\times 10^{-2}$ )
RTS-EM [12]	9.86	3.54
PS-EM [7]	8.54	9.01
RBPS-EM [this paper]	9.77	3.47

The key difference between PS-EM and RBPS-EM is that in the former, the particles have to cover the distribution in two dimensions. In RBPS-EM we marginalize one of the dimensions analytically and thus only need to deal with one of the dimensions using particles. In this example, it is clear that 50 particles is insufficient to give an accurate enough

approximation of the distribution in two dimensions. In [7], PS-EM is shown to have good performance even with few particles. However, this is only true for low order systems. To deal with higher order systems we need to increase the number of particles, resulting in an increased computational complexity. As the dimension goes up, this will soon make standard particle method inapplicable.

### B. A Nonlinear Example

We shall now study a fourth order nonlinear system, where three of the states are conditionally linear Gaussian,

$$a_{t+1} = \theta_1 \arctan a_t + (\theta_2 \ 0 \ 0) z_t + w_{a,t}, \quad (35a)$$

$$z_{t+1} = \begin{pmatrix} 1 & \theta_3 & 0 \\ 0 & \theta_4 \cos(\theta_5) & -\theta_4 \sin(\theta_5) \\ 0 & \theta_4 \sin(\theta_5) & \theta_4 \cos(\theta_5) \end{pmatrix} z_t + w_{z,t}, \quad (35b)$$

$$y_t = \begin{pmatrix} 0.1 a_t^2 \operatorname{sgn}(a_t) \\ 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 \\ 1 & -1 & 1 \end{pmatrix} z_t + e_t, \quad (35c)$$

with  $Q = 0.01I_{4 \times 4}$ ,  $R = 0.1I_{2 \times 2}$ ,  $a_1 \sim \mathcal{N}(0, 1)$ ,  $z_1 = (0 \ 0 \ 0)^T$  and the parameters  $\theta = \{\theta_i\}_{i=1}^5$ . The true parameter vector is  $\theta^* = (1 \ 1 \ 0.3 \ 0.968 \ 0.315)^T$ . The  $z$ -system is oscillatory, with poles in 1,  $0.92 \pm 0.3i$  and the  $z$ -variables are connected to the nonlinear  $a$ -system through  $z_{1,t}$ . First, we assume that the  $z$ -system and also the parameter  $\theta_1$  are known, i.e. we are only concerned with finding the parameter  $\theta_2$  connecting the two systems.

For this Monte Carlo study, 130 realizations of data  $Y_N$  were generated, each consisting of  $N = 200$  samples. The parameter was thereafter identified using RBPS-EM and PS-EM, respectively. Both methods used  $M = 50$  particles and 500 iterations of the EM algorithm. The initial parameter estimate was chosen randomly in the interval  $[0, 2]$  for each simulation. Figure 1 illustrates the convergence of the two methods. For RBPS-EM the Monte Carlo mean and the standard deviation of the final parameter estimate was  $\hat{\theta}_{500}^{\text{RBPS}} = 0.991 \pm 0.059$ . For PS-EM the corresponding figures were  $\hat{\theta}_{500}^{\text{PS}} = 1.048 \pm 0.162$ . The parameter variance is much higher for PS-EM than for RBPS-EM, which is obvious from Figure 1 as well. When using Rao-Blackwellization we know that the variance of the estimated states will be reduced [21]. The intuition is that this will influence the variance of the estimated parameters as well, and from the above results it seems as if this truly is the case.

As a final experiment we assumed all parameters  $\theta = \{\theta_i\}_{i=1}^5$  to be unknown. PS-EM and RBPS-EM were run on 70 realizations of data, for 1000 iterations of the EM algorithm ( $N = 200$  and  $M = 50$  as before). Parameters  $\theta_1$ ,  $\theta_2$  and  $\theta_3$  were initialized randomly in the intervals  $\pm 100\%$  of their nominal values. The initial value for  $\theta_4$  was chosen randomly in the interval  $[0, 1]$  and for  $\theta_5$  in the interval  $[0, \pi/2]$  (i.e. the complex poles of the  $z$ -system were initiated randomly in the first and the fourth quadrants of the unit circle). In one of the 70 experiments, the estimates diverged. This experiment has been removed from the comparison, not to overshadow the standard deviations. Table II summarizes the results.

The difference between the two methods becomes even more evident in this, more challenging, experiment. The

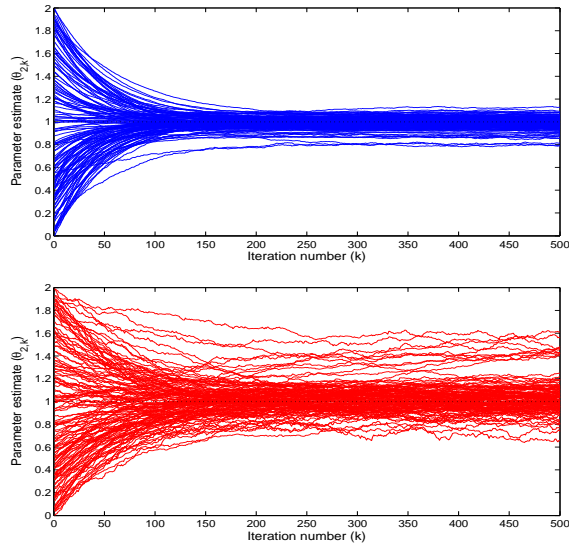


Fig. 1. Parameter estimates as functions of iteration number for RBPS-EM (top) and PS-EM (bottom). Each line corresponds to one realization of data. The true parameter value is 1.

TABLE II  
MONTE CARLO MEANS AND STANDARD DEVIATIONS

Parameter	True value	RBPS-EM	PS-EM
$\theta_1$	1	$0.966 \pm 0.163$	$0.965 \pm 0.335$
$\theta_2$	1	$1.053 \pm 0.163$	$1.067 \pm 0.336$
$\theta_3$	0.3	$0.295 \pm 0.094$	$-0.754 \pm 1.141$
$\theta_4$	0.968	$0.967 \pm 0.015$	$0.880 \pm 0.154$
$\theta_5$	0.315	$0.309 \pm 0.057$	$0.064 \pm 0.285$

parameter standard deviations for PS-EM are much higher for all parameters, and for two ( $\theta_3$  and  $\theta_5$ ) of the five parameters, PS-EM totally fails in identifying the true values.

## VI. CONCLUSION

In this paper we have presented a new method for maximum likelihood parameter estimation in nonlinear state-space models containing conditionally linear Gaussian substructures. The method is based on the expectation maximization algorithm and a Rao-Blackwellized Particle Smoother (RBPS). As a secondary contribution we have extended a previously existing RBPS [13], to be able to handle the fully interconnected mixed model under study.

Through simulations, the proposed method is shown to reduce the variance of the parameter estimate, when compared to a similar method based on standard particle methods [7]. It is well known that the variance of the state estimates is reduced when Rao-Blackwellization is used. Here, we have shown that this seems to be the case also for the estimated parameters. Furthermore, we have shown that we are able to estimate parameters in a special type of high-dimensional nonlinear state-space models, for which standard particle methods fail due to the high state dimension.

## REFERENCES

- [1] L. Ljung and A. Vicino, Eds., *Special Issue on System Identification. IEEE Transactions on Automatic Control*, 2005, vol. 50 (10).
- [2] L. Ljung, "Perspectives on system identification," in *Proceedings of the 17th IFAC World Congress*, Seoul, South Korea, Jul 2008, pp. 7172–7184, plenary lecture.
- [3] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*. New York, USA: Springer Verlag, 2001.
- [4] A. Doucet and A. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," in *Handbook of Nonlinear Filtering (to appear)*. Oxford University Press, 2010.
- [5] C. Andrieu, A. Doucet, S. S. Singh, and V. B. Tadić, "Particle methods for change detection, system identification, and control," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 423–438, Mar. 2004.
- [6] N. Kantas, A. Doucet, S. Singh, and J. Maciejowski, "An overview of sequential Monte Carlo methods for parameter estimation in general state-space models," in *Proceedings of the 15th IFAC Symposium on System Identification*, Saint-Malo, France, Jul. 2009, pp. 774–785.
- [7] T. B. Schön, A. Wills, and B. Ninness, "System identification of nonlinear state-space models," *Provisionally accepted to Automatica*, 2010.
- [8] J. Olsson, R. Douc, O. Cappé, and E. Moulines, "Sequential Monte Carlo smoothing with application to parameter estimation in nonlinear state-space models," *Bernoulli*, vol. 14, no. 1, pp. 155–179, 2008.
- [9] R. B. Gopaluni, "Identification of nonlinear processes with known model structure using missing observations," in *Proceedings of the 17th IFAC World Congress*, Seoul, South Korea, Jul. 2008.
- [10] G. Poyiadjis, A. Doucet, and S. Singh, "Sequential monte carlo computation of the score and observed information matrix in state-space models with application to parameter estimation," Cambridge University Engineering Department, Cambridge, UK, Tech. Rep. CUED/F-INFENG/TR 628, May 2009.
- [11] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1–38, 1977.
- [12] S. Gibson and B. Ninness, "Robust maximum-likelihood estimation of multivariable dynamic systems," *Automatica*, vol. 41, no. 10, pp. 1667–1682, 2005.
- [13] W. Fong, S. J. Godsill, A. Doucet, and M. West, "Monte Carlo smoothing with application to audio signal enhancement," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 438–449, Feb. 2002.
- [14] M. Briers, A. Doucet, and S. Maskell, "Smoothing algorithms for state-space models," *Annals of the Institute of Statistical Mathematics*, vol. 62, no. 1, pp. 61–89, Feb. 2010.
- [15] M. K. Pitt, "Smooth particle filters for likelihood evaluation and maximisation," Department of economics, University of Warwick, Coventry, UK, Tech. Rep. Warwick economic research papers No 651, Jul. 2002.
- [16] G. Poyiadjis, A. Doucet, and S. S. Singh, "Maximum likelihood parameter estimation in general state-space models using particle methods," in *Proceedings of the American Statistical Association*, Minneapolis, USA, Aug. 2005.
- [17] G. Kitagawa, "A self-organizing state-space model," *Journal of the American Statistical Association*, vol. 93, no. 443, pp. 1203–1215, Sep. 1998.
- [18] T. Schön and F. Gustafsson, "Particle filters for system identification of state-space models linear in either parameters or states," in *Proceedings of the 13th IFAC Symposium on System Identification (SYSID)*, Rotterdam, The Netherlands, Sep. 2003, pp. 1287–1292.
- [19] T. B. Schön, "An explanation of the expectation maximization algorithm," Division of Automatic Control, Linköping University, Linköping, Sweden, Tech. Rep. LiTH-ISY-R-2915, Aug. 2009.
- [20] J. Nocedal and S. J. Wright, *Numerical Optimization*, ser. Operations Research. New York, USA: Springer, 2000.
- [21] A. Doucet, N. de Freitas, K. Murphy, and S. Russell, "Rao-blackwellised particle filtering for dynamic bayesian networks," in *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, 2000, pp. 176–183.
- [22] F. Lindsten and T. B. Schön, "Inference in mixed linear/nonlinear state-space models using sequential Monte Carlo," Department of Electrical Engineering, Linköping University, Linköping, Sweden, Tech. Rep. LiTH-ISY-R-2946, March 2010. [Online]. Available: <http://www.control.isy.liu.se/research/reports/2010/2946.pdf>
- [23] T. B. Schön, F. Gustafsson, and P.-J. Nordlund, "Marginalized particle filters for mixed linear/nonlinear state-space models," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2279–2289, Jul. 2005.
- [24] P. Fearnhead, D. Wyncoll, and J. Tawn, "A sequential smoothing algorithm with linear computational cost," Preprint, May 2008.
- [25] R. Douc, A. Garivier, E. Moulines, and J. Olsson, "Sequential Monte Carlo smoothing for general state space hidden Markov models," *Submitted to Annals of Applied Probability*, 2010.
- [26] H. E. Rauch, F. Tung, and C. T. Striebel, "Maximum likelihood estimates of linear dynamic systems," *AIAA Journal*, vol. 3, no. 8, pp. 1445–1450, Aug. 1965.