UNSUPERVISED SIGNAL RESTORATION IN PARTIALLY OBSERVED MARKOV CHAINS

Boujemaa Ait El Fquih and François Desbouvries

GET / INT / dépt. CITI and CNRS UMR 5157
9 rue Charles Fourier, 91011 Evry, France
Boujemaa.Ait_elfquih@int-evry.fr, François.Desbouvries@int-evry.fr

ABSTRACT

An important problem in signal processing consists in estimating an unobservable process $\mathbf{x} = \{\mathbf{x}_n\}_{n\in\mathbb{N}}$ from an observed process $\mathbf{y} = \{\mathbf{y}_n\}_{n\in\mathbb{N}}$. In Linear Gaussian Hidden Markov Chains (LGHMC), recursive solutions are given by Kalman-like Bayesian restoration algorithms. In this paper, we consider the more general framework of Linear Gaussian Triplet Markov Chains (LGTMC), i.e. of models in which the triplet $(\mathbf{x}, \mathbf{r}, \mathbf{y})$ (where $\mathbf{r} = \{\mathbf{r}_n\}_{n\in\mathbb{N}}$ is some additional process) is Markovian and Gaussian. We address unsupervised restoration in LGTMC by extending to LGTMC the EM parameter estimation algorithm which was already developed in classical state-space models.

1. INTRODUCTION

An important problem in signal processing consists in recursively estimating an unobservable process $\mathbf{x} = \{\mathbf{x}_n\}_{n\in\mathbb{N}}$ from an observed process $\mathbf{y} = \{\mathbf{y}_n\}_{n\in\mathbb{N}}$. This is done classically in the framework of Hidden Markov Chains (HMC), which have been extensively studied for many years (see e.g. the recent tutorial [1]).

In this paper we deal with the recently introduced Pairwise [2] (PMC) and Triplet [3] Markov Chains (TMC). In the PMC model we assume that the pair $(\mathbf{x}, \mathbf{y})$ is a Markov Chain (MC), and in the TMC model that the triplet $(\mathbf{x}, \mathbf{r}, \mathbf{y})$ (in which $\mathbf{r} = \{\mathbf{r}_n\}_{n\in\mathbb{N}}$ is some additional process) is an MC. These models are more general than the HMC model and yet enable the development of efficient restoration algorithms of the hidden process $\mathbf{x}$. In particular, a Kalman-like filtering algorithm for TMC has been proposed in [4] [5], and smoothing algorithms for TMC have been proposed in [6].

Now, in [4]-[6] the parameters were assumed to be known. In this paper we thus address the unsupervised case and develop the EM parameter estimation algorithm for linear Gaussian TMC. The rest of this paper is organized as follows. In section 2 we briefly recall the three embedded HMC, PMC and TMC models. In section 3 we develop the EM algorithm for linear Gaussian TMC. Finally in section 4 we perform some simulations and compare the restoration results in both the supervised and unsupervised cases.

2. PARTIALLY OBSERVED MARKOV CHAINS

2.1. HMC and Kalman filtering

HMC are widely used in such topics as speech recognition, digital communications, tracking and control. In an HMC, $\mathbf{x}$ is first assumed to be an MC (by the very meaning of the words "HMC"), and next the stochastic interactions of $\mathbf{x}$ and $\mathbf{y}$ are designed in such a way that $\mathbf{x}$ can be efficiently restored from $\mathbf{y}$. Let us for instance consider the classical state-space system :

$$\left\{ \begin{array}{lcl} \mathbf{x}_{n+1} & = & \mathbf{F}_n\mathbf{x}_n + \mathbf{G}_n\mathbf{u}_n \\ \mathbf{y}_n & = & \mathbf{H}_n\mathbf{x}_n + \mathbf{J}_n\mathbf{v}_n \end{array} \right. , \tag{1}$$

in which $\mathbf{x}_n \in \mathbb{R}^{n_\mathbf{x}}$ is the state, $\mathbf{y}_n \in \mathbb{R}^{n_\mathbf{y}}$ is the observation, $\mathbf{u}_n \in \mathbb{R}^{n_\mathbf{u}}$ is the process noise and $\mathbf{v}_n \in \mathbb{R}^{n_\mathbf{v}}$ is the measurement noise. Let $\mathbf{x}_{0:n} = \{\mathbf{x}_i\}_{i=0}^n$ and $\mathbf{y}_{0:n} = \{\mathbf{y}_i\}_{i=0}^n$. Let also $p(\mathbf{x}_n)$, $p(\mathbf{x}_{0:n})$ and $p(\mathbf{x}_n|\mathbf{y}_{0:n})$, say, denote the probability density function (pdf) (w.r.t. Lebesgue measure) of $\mathbf{x}_n$, the pdf of $\mathbf{x}_{0:n}$, and the pdf of $\mathbf{x}_n$, conditional on $\mathbf{y}_{0:n}$, respectively; the other pdf are defined similarly. The processes $\mathbf{u} = \{\mathbf{u}_n\}_{n\in\mathbb{N}}$ and $\mathbf{v} = \{\mathbf{v}_n\}_{n\in\mathbb{N}}$ are assumed to be independent, jointly independent and independent of $\mathbf{x}_0$. As a consequence,

$$p(\mathbf{x}_{n+1}|\mathbf{x}_{0:n}) = p(\mathbf{x}_{n+1}|\mathbf{x}_n) ; \tag{2}$$

$$p(\mathbf{y}_{0:n}|\mathbf{x}_{0:n}) = \prod_{i=0}^{n} p(\mathbf{y}_i|\mathbf{x}_{0:n}) ; \tag{3}$$

$$p(\mathbf{y}_i|\mathbf{x}_{0:n}) = p(\mathbf{y}_i|\mathbf{x}_i) \text{ for all } i,\ 0 \le i \le n . \tag{4}$$

In other words, $\mathbf{x}$ is an MC, and since $\mathbf{x}$ is known only through the observed process $\mathbf{y}$, (1) is an HMC (with continuous state-space).

The filtering problem consists in computing the posterior pdf $p(\mathbf{x}_n|\mathbf{y}_{0:n})$. If furthermore $\mathbf{x}_0$ and $\mathbf{w}_n = (\mathbf{u}_n, \mathbf{v}_n)$ are Gaussian, then $p(\mathbf{x}_n|\mathbf{y}_{0:n})$ is also Gaussian and is thus described by its mean and covariance matrix. Propagating $p(\mathbf{x}_n|\mathbf{y}_{0:n})$ amounts to propagating its parameters, and the general recursive algorithm for computing $p(\mathbf{x}_n|\mathbf{y}_{0:n})$ reduces to the celebrated Kalman filter.

2.2. Embedded Markovian models : HMC $\subset$ PMC $\subset$ TMC

Let us now call PMC a model in which the pair $(\mathbf{x}, \mathbf{y})$ is assumed to be an MC. So in a PMC $\mathbf{x}$ and $\mathbf{y}$ are modeled altogether (and in a symmetric way), and a PMC can indeed be seen as a partially observed vector MC, in which we observe one component $\mathbf{y}$ and we want to restore the other one $\mathbf{x}$.

Our interest for PMC comes from the following observation. If (2) to (4) hold, then $(\mathbf{x}, \mathbf{y})$ is an MC, so any HMC is also a PMC. The converse is not true, because if $(\mathbf{x}, \mathbf{y})$ is a (vector) MC then the marginal process $\mathbf{x}$ is not necessarily an MC; moreover, conditionnally on $\mathbf{x}_{0:n}$, the variables $\{\mathbf{y}_i\}_{i=0}^n$ form an MC and thus are not necessarily independent [4]. On the other hand, due to the symmetry of the PMC model, the conditional law of $\mathbf{x}_{0:n}$ given $\mathbf{y}_{0:n}$ is also Markovian. This key computational property (which in the context of HMC is well known, see e.g. [1, eq. (5.21) p. 1539]), in turn, enables the derivation of efficient HMC-like restoration algorithms. In particular, in the linear Gaussian case, the extension to PMC of the Kalman filter has been considered in [4] [5].

The PMC model can be further generalized to the TMC model [3] which we now recall. A TMC is a stochastic dynamical model which describes the interactions between 3 processes : the hidden process $\mathbf{x}$, the observed process $\mathbf{y}$, and a third process $\mathbf{r}$ which, depending on the application, can have different physical meanings. By definition, the triplet $\mathbf{t} = (\mathbf{x}, \mathbf{r}, \mathbf{y})$ is a TMC if $(\mathbf{x}, \mathbf{r}, \mathbf{y})$ is a (vector) Markov chain. The interest of TMC is twofold :

- As far as modeling is concerned, if $(\mathbf{r}, (\mathbf{x}, \mathbf{y}))$ is an MC then the marginal process $(\mathbf{x}, \mathbf{y})$ is not necessarily an MC, so TMC are not necessarily PMC;
- As far as restoration is concerned, the TMC $(\mathbf{r}, \mathbf{x}, \mathbf{y})$ can be viewed as the PMC $((\mathbf{r}, \mathbf{x}), \mathbf{y})$; so $\mathbf{x}^* = (\mathbf{r}, \mathbf{x})$ can be restored from $\mathbf{y}$ by a PMC algorithm, and finally $\mathbf{x}$ is obtained by marginalization (such algorithms have been proposed in the discrete [3] or linear Gaussian [5] cases).

It happens that TMC include some classical generalizations of model (1), such as jump-Markov state-space systems, or state-space systems with colored process and / or measurement noises [5] (other examples of TMC can be found in [7]). Finally, let us notice that in practice computer experiments have demonstrated the superiority of PMC [8] (resp. TMC [9]) over HMC in the context of image segmentation.

3. PARAMETER ESTIMATION IN LINEAR GAUSSIAN TMC

The aim of this section is to derive and implement the EM parameter estimation algorithm in constant parameter linear Gaussian TMC. Let $\mathbf{x}^*_n = [\mathbf{x}_n^T, \mathbf{r}_n^T]^T$. From now on we shall thus assume that

$$\underbrace{\begin{bmatrix} \mathbf{x}^*_{n+1} \\ \mathbf{y}_n \end{bmatrix}}_{\mathbf{t}_{n+1}} = \underbrace{\begin{bmatrix} \mathcal{F}^{\mathbf{x}^*,\mathbf{x}^*} & \mathcal{F}^{\mathbf{x}^*,\mathbf{y}} \\ \mathcal{F}^{\mathbf{y},\mathbf{x}^*} & \mathcal{F}^{\mathbf{y},\mathbf{y}} \end{bmatrix}}_{\mathcal{F}} \begin{bmatrix} \mathbf{x}^*_n \\ \mathbf{y}_{n-1} \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{w}_n^{\mathbf{x}^*} \\ \mathbf{w}_n^{\mathbf{y}} \end{bmatrix}}_{\mathbf{w}_n}, \quad (5)$$

in which $\mathbf{w} = \{\mathbf{w}_n\}_{n\in\mathbb{N}}$ is independent and independent of $\mathbf{t}_0 = \mathbf{x}^*_0$, $\mathbf{x}^*_0 \sim \mathcal{N}(\widehat{\mathbf{x}}^*_0, \mathbf{P}^*_0)$ and

$$\mathbf{w}_n \sim \mathcal{N}(\mathbf{0}, \underbrace{\begin{bmatrix} \mathcal{Q}^{\mathbf{x}^*,\mathbf{x}^*} & \mathcal{Q}^{\mathbf{x}^*,\mathbf{y}} \\ \mathcal{Q}^{\mathbf{y},\mathbf{x}^*} & \mathcal{Q}^{\mathbf{y},\mathbf{y}} \end{bmatrix}}_{\mathcal{Q}}). \quad (6)$$

Then $\mathbf{t} = \{\mathbf{t}_n\}_{n\in\mathbb{N}}$ is a Gauss-Markov process and we set $p(\mathbf{x}^*_i|\mathbf{y}_{0:j}) \sim \mathcal{N}(\widehat{\mathbf{x}}^*_{i|j}, \mathbf{P}^*_{i|j})$. We also set $\mathbf{t}_{0:n} = \{\mathbf{t}_i\}_{i=0}^n$ and we assume that $\mathbf{P}^*_0$, $\mathcal{Q}$ and $\mathcal{Q}^{\mathbf{y},\mathbf{y}}$ are invertible.

3.1. EM algorithm

Let $\Theta = (\Theta_0, \Theta_1)$, with $\Theta_0 = (\widehat{\mathbf{x}}^*_0, \mathbf{P}^*_0)$ and $\Theta_1 = (\mathcal{F}, \mathcal{Q})$, be the parameters in model (5). We want to compute the EM algorithm [10] (see also [11] [12] [13] and the references therein), which consists in the recursion

$$\Theta^{(i)} = \arg\max_{\Theta} q(\Theta^{(i-1)}, \Theta), \quad (7)$$

in which

$$q(\Theta^{(i-1)}, \Theta) = E_{\Theta^{(i-1)}}(\ln p_\Theta(\mathbf{t}_{0:N+1})|\mathbf{y}_{0:N}) \quad (8)$$
$$= \int (\ln p_\Theta(\mathbf{t}_{0:N+1})) p_{\Theta^{(i-1)}}(\mathbf{x}^*_{0:N+1}|\mathbf{y}_{0:N}) d\mathbf{x}^*_{0:N+1}.$$

Since $\mathbf{t}$ is a TMC, the joint pdf $p_\Theta(\mathbf{t}_{0:N+1})$ factorizes as

$$p_\Theta(\mathbf{t}_{0:N+1}) = p_{\Theta_0}(\mathbf{x}^*_0) p_{\Theta_1}(\mathbf{t}_1|\mathbf{x}^*_0) \prod_{n=1}^{N} p_{\Theta_1}(\mathbf{t}_{n+1}|\mathbf{t}_n)$$

with

$$p_{\Theta_0}(\mathbf{x}^*_0) \sim \mathcal{N}(\widehat{\mathbf{x}}^*_0, \mathbf{P}^*_0),$$
$$p_{\Theta_1}(\mathbf{t}_1|\mathbf{x}^*_0) \sim \mathcal{N}(\mathcal{F}\mathcal{J}\mathbf{x}^*_0, \mathcal{Q}),$$
$$p_{\Theta_1}(\mathbf{t}_{n+1}|\mathbf{t}_n) \sim \mathcal{N}(\mathcal{F}\mathbf{t}_n, \mathcal{Q})$$

and $\mathcal{J} = [\, \mathbf{I}_{n_{\mathbf{x}^*} \times n_{\mathbf{x}^*}}, \mathbf{0}^T_{n_{\mathbf{y}} \times n_{\mathbf{x}^*}}]^T$. Droping the explicit dependence on the current candidate parameter $\Theta^{(i-1)}$, the q-function thus decouples as

$$q(\Theta) = q_{-1}(\Theta_0) + q_0(\Theta_1) + \sum_{n=1}^{N} q_n(\Theta_1). \quad (9)$$

3.1.1. E-step

Let us address the E-step. Let us first take $n \geq 1$. One can show that

$$
\begin{aligned}
q_n(\Theta_1) &= E_{\Theta^{(i-1)}}(\ln p_{\Theta_1}(\mathbf{t}_{n+1}|\mathbf{t}_n)|\mathbf{y}_{0:N}) \\
&= \int (\ln p_{\Theta_1}(\mathbf{t}_{n+1}|\mathbf{t}_n)) p_{\Theta^{(i-1)}}(\mathbf{x}_n^*, \mathbf{x}_{n+1}^*|\mathbf{y}_{0:N}) d\mathbf{x}_n^* d\mathbf{x}_{n+1}^* \\
&= \frac{-1}{2}(n_{\mathbf{t}} \ln(2\pi) - \ln|\mathcal{Q}^{-1}| - \text{trace}[\mathcal{F}^T \mathcal{Q}^{-1} \mathbf{C}_{\mathbf{t}_{n+1},\mathbf{t}_n}] \\
&\quad - \text{trace}[\mathcal{Q}^{-1} \mathcal{F} \mathbf{C}_{\mathbf{t}_{n+1},\mathbf{t}_n}^T] + \text{trace}[\mathcal{Q}^{-1} \mathbf{C}_{\mathbf{t}_{n+1},\mathbf{t}_{n+1}}] \\
&\quad + \text{trace}[\mathcal{F}^T \mathcal{Q}^{-1} \mathcal{F} \mathbf{C}_{\mathbf{t}_n,\mathbf{t}_n}]),
\end{aligned}
$$

in which $|\mathcal{Q}^{-1}|$ denotes the determinant of $\mathcal{Q}^{-1}$, and

$$
\begin{aligned}
\mathbf{C}_{\mathbf{t}_n,\mathbf{t}_n} &= E(\mathbf{t}_n \mathbf{t}_n^T|\mathbf{y}_{0:N}) = \begin{bmatrix} \mathbf{P}_{n|N}^* & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} + \begin{bmatrix} \widehat{\mathbf{x}}_{n|N}^* \\ \mathbf{y}_{n-1} \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{x}}_{n|N}^* \\ \mathbf{y}_{n-1} \end{bmatrix}^T, \\
\mathbf{C}_{\mathbf{t}_{n+1},\mathbf{t}_n} &= \begin{bmatrix} \text{Cov}(\mathbf{x}_{n+1}^*, \mathbf{x}_n^*|\mathbf{y}_{0:N}) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} + \begin{bmatrix} \widehat{\mathbf{x}}_{n+1|N}^* \\ \mathbf{y}_n \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{x}}_{n|N}^* \\ \mathbf{y}_{n-1} \end{bmatrix}^T.
\end{aligned}
$$

Let us now compute $q_{-1}(\Theta_0)$ and $q_0(\Theta_1)$. We have

$$
\begin{aligned}
q_{-1}(\Theta_0) &= \frac{-1}{2}(n_{\mathbf{x}^*} \ln(2\pi) - \ln|\mathbf{P}_0^{*-1}| + \text{trace}(\mathbf{P}_0^{*-1} \mathbf{P}_{0|N}^*) \\
&\quad + (\widehat{\mathbf{x}}_{0|N}^* - \widehat{\mathbf{x}}_0^*)^T \mathbf{P}_0^{*-1} (\widehat{\mathbf{x}}_{0|N}^* - \widehat{\mathbf{x}}_0^*)), \\
q_0(\Theta_1) &= \frac{-1}{2}(n_{\mathbf{t}} \ln(2\pi) - \ln|\mathcal{Q}^{-1}| - \text{trace}[\mathcal{J}^T \mathcal{F}^T \mathcal{Q}^{-1} \mathbf{C}_{\mathbf{t}_1,\mathbf{x}_0^*}] \\
&\quad - \text{trace}[\mathcal{Q}^{-1} \mathcal{F} \mathcal{J} \mathbf{C}_{\mathbf{t}_1,\mathbf{x}_0^*}^T] + \text{trace}[\mathcal{Q}^{-1} \mathbf{C}_{\mathbf{t}_1,\mathbf{t}_1}] \\
&\quad + \text{trace}[\mathcal{J}^T \mathcal{F}^T \mathcal{Q}^{-1} \mathcal{F} \mathcal{J} \mathbf{C}_{\mathbf{x}_0^*,\mathbf{x}_0^*}]),
\end{aligned}
$$

with $\mathbf{C}_{\mathbf{x}_0^*,\mathbf{x}_0^*} = \mathbf{P}_{0|N}^* + \widehat{\mathbf{x}}_{0|N}^* (\widehat{\mathbf{x}}_{0|N}^*)^T$ and

$$
\mathbf{C}_{\mathbf{t}_1,\mathbf{x}_0^*} = \begin{bmatrix} \text{Cov}(\mathbf{x}_1^*, \mathbf{x}_0^*|\mathbf{y}_{0:N}) \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \widehat{\mathbf{x}}_{1|N}^* \\ \mathbf{y}_0 \end{bmatrix} (\widehat{\mathbf{x}}_{0|N}^*)^T. \quad (10)
$$

3.1.2. M-step

We now address the M-step, i.e. the maximization w.r.t. Θ of (9). Using standard matrix tools [14] [15] we get

$$
\begin{aligned}
\widehat{\mathbf{x}}_0^{*(i)} &= \widehat{\mathbf{x}}_{0|N}^*, && (11) \\
\mathbf{P}_0^{*(i)} &= \mathbf{P}_{0|N}^*, && (12) \\
\mathcal{F}^{(i)} &= \widetilde{\mathbf{C}}_{\mathbf{t}_{n+1},\mathbf{t}_n} \widetilde{\mathbf{C}}_{\mathbf{t}_n,\mathbf{t}_n}^{-1}, && (13) \\
\mathcal{Q}^{(i)} &= \frac{1}{N+1}[\widetilde{\mathbf{C}}_{\mathbf{t}_{n+1},\mathbf{t}_{n+1}} - \widetilde{\mathbf{C}}_{\mathbf{t}_{n+1},\mathbf{t}_n} \widetilde{\mathbf{C}}_{\mathbf{t}_n,\mathbf{t}_n}^{-1} \widetilde{\mathbf{C}}_{\mathbf{t}_{n+1},\mathbf{t}_n}^T], && (14)
\end{aligned}
$$

in which

$$
\begin{aligned}
\widetilde{\mathbf{C}}_{\mathbf{t}_{n+1},\mathbf{t}_n} &= \mathbf{C}_{\mathbf{t}_1,\mathbf{x}_0^*} \mathcal{J}^T + \sum_{n=1}^{N} \mathbf{C}_{\mathbf{t}_{n+1},\mathbf{t}_n}, && (15) \\
\widetilde{\mathbf{C}}_{\mathbf{t}_n,\mathbf{t}_n} &= \mathcal{J} \mathbf{C}_{\mathbf{x}_0^*,\mathbf{x}_0^*} \mathcal{J}^T + \sum_{n=1}^{N} \mathbf{C}_{\mathbf{t}_n,\mathbf{t}_n}, && (16) \\
\widetilde{\mathbf{C}}_{\mathbf{t}_{n+1},\mathbf{t}_{n+1}} &= \sum_{n=0}^{N} \mathbf{C}_{\mathbf{t}_{n+1},\mathbf{t}_{n+1}}. && (17)
\end{aligned}
$$

3.2. Practical implementation

In order to implement the recursion $(\Theta^{(i-1)} \rightarrow \Theta^{(i)})$, we see from (11) (12) (13) (14) that we need to compute $\widehat{\mathbf{x}}_{n|N}^*$, $\mathbf{P}_{n|N}^*$ and $\text{Cov}(\mathbf{x}_{n+1}^*, \mathbf{x}_n^*|\mathbf{y}_{0:N})$ for all n.

First, $\widehat{\mathbf{x}}_{n|N}^*$ and $\mathbf{P}_{n|N}^*$ are given for instance by the RTS smoothing algorithm for TMC [6, Proposition 3] (or alternately by the TMC Two-Filter algorithm [6, Proposition 4]), which we recall for convenience of the reader :

$$
\begin{aligned}
\mathbf{K}_{n|N}^* &= \mathbf{P}_{n|n}^*[\mathcal{F}^{\mathbf{x}^*,\mathbf{x}^*} \\
&\quad - \mathcal{Q}^{\mathbf{x}^*,\mathbf{y}}(\mathcal{Q}^{\mathbf{y},\mathbf{y}})^{-1} \mathcal{F}^{\mathbf{y},\mathbf{x}^*}]^T \mathbf{P}_{n+1|n}^{*-1}, && (18) \\
\widehat{\mathbf{x}}_{n|N}^* &= \widehat{\mathbf{x}}_{n|n}^* + \mathbf{K}_{n|N}^*[\widehat{\mathbf{x}}_{n+1|N}^* - \widehat{\mathbf{x}}_{n+1|n}^*], && (19) \\
\mathbf{P}_{n|N}^* &= \mathbf{P}_{n|n}^* - \mathbf{K}_{n|N}^* \mathbf{P}_{n+1|n}^* \mathbf{K}_{n|N}^{*T} \\
&\quad + \mathbf{K}_{n|N}^* \mathbf{P}_{n+1|N}^* \mathbf{K}_{n|N}^{*T}. && (20)
\end{aligned}
$$

So, we see from these equations that $(\widehat{\mathbf{x}}_{n|N}^*, \mathbf{P}_{n|N}^*)$ can be computed recursively (in the backward direction) provided $(\widehat{\mathbf{x}}_{n|n}^*, \mathbf{P}_{n|n}^*)$ and $(\widehat{\mathbf{x}}_{n+1|n}^*, \mathbf{P}_{n+1|n}^*)$ are known; these, in turn, can be computed recursively (in the forward direction) by the TMC Kalman filter algorithm, see [4] [5]. On the other hand, $\text{Cov}(\mathbf{x}_{n+1}^*, \mathbf{x}_n^*|\mathbf{y}_{0:N})$ is not directly given by the TMC RTS smoother, but can be computed as

$$
\text{Cov}(\mathbf{x}_{n+1}^*, \mathbf{x}_n^*|\mathbf{y}_{0:N}) = \mathbf{P}_{n+1|N}^* \mathbf{K}_{n|N}^{*T}. \quad (21)
$$

Proof 1

We have [6]

$$
p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*, \mathbf{y}_{0:N}) = p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*, \mathbf{y}_{0:n})
$$

$$
\sim \mathcal{N}(\widehat{\mathbf{x}}_{n|n}^* + \mathbf{K}_{n|N}^*(\mathbf{x}_{n+1}^* - \widehat{\mathbf{x}}_{n+1|n}^*), \mathbf{P}_{n|n}^* - \mathbf{K}_{n|N}^* \mathbf{P}_{n+1|N}^* \mathbf{K}_{n|N}^{*T}).
$$

On the other hand, $p(\mathbf{x}_{n+1}^*|\mathbf{y}_{0:N}) \sim \mathcal{N}(\widehat{\mathbf{x}}_{n+1|N}^*, \mathbf{P}_{n+1|N}^*)$. So

$$
p(\mathbf{x}_n^*, \mathbf{x}_{n+1}^*|\mathbf{y}_{0:N}) \sim \mathcal{N}\left(\begin{bmatrix} \widehat{\mathbf{x}}_{n|N}^* \\ \widehat{\mathbf{x}}_{n+1|N}^* \end{bmatrix}, \begin{bmatrix} \mathbf{P}_{n|N}^* & \mathbf{K}_{n|N}^* \mathbf{P}_{n+1|N}^* \\ \mathbf{P}_{n+1|N}^* \mathbf{K}_{n|N}^{*T} & \mathbf{P}_{n+1|N}^* \end{bmatrix}\right),
$$

whence (21). ■

Let us finally summarize our estimation algorithm.

Expectation-Maximization algorithm

1. Initialization of the parameters.

   Choose $\widehat{\mathbf{x}}_0^{*(0)}$, $\mathbf{P}_0^{*(0)}$, $\mathcal{F}^{(0)}$ and $\mathcal{Q}^{(0)}$.

2. Iteration $(i-1) \rightarrow (i)$.

   Run a TMC smoother with parameters estimates $\Theta^{(i-1)} = (\widehat{\mathbf{x}}_0^{*(i-1)}, \mathbf{P}_0^{*(i-1)}, \mathcal{F}^{(i-1)}, \mathcal{Q}^{(i-1)})$, compute (19), (20) and (21), and finally compute $\Theta^{(i)}$ from (11), (12), (13) and (14).

4. SIMULATIONS

In this final section we perform some simulations. We consider a linear Gaussian TMC model with parameters $\widehat{\mathbf{x}}_0^* = [.5, .5]^T$, $\mathbf{P}_0^* = 2.5\mathbf{I}_2$, and

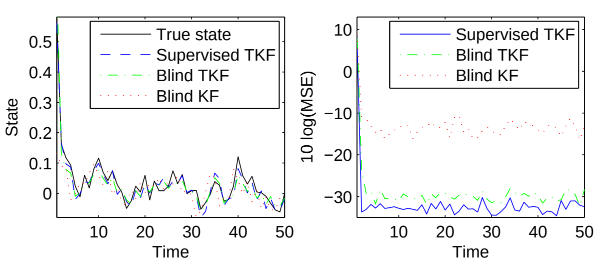
$$\mathcal{F} = \begin{bmatrix} .12 & .10 & .11 \\ .11 & .10 & .12 \\ .10 & .11 & .12 \end{bmatrix}, \quad \mathcal{Q} = \begin{bmatrix} .18 & .15 & .16 \\ .15 & .18 & .14 \\ .16 & .14 & .18 \end{bmatrix},$$

and we restore the hidden process x by 3 different algorithms. The first figure illustrates the restored state, and the second one the empirical MSE (in dB).

In the first two experiments, we use a TMC Kalman filter, respectively in a supervised and unsupervised environment; in the blind framework, the model parameters are first estimated by the EM algorithm initialized by $\widehat{\mathbf{x}}_0^{*(0)} = [.6, .6]^T$, $\mathbf{P}_0^{*(0)} = 3\mathbf{I}_2$, and

$$\mathcal{F}^{(0)} = \begin{bmatrix} .10 & .09 & .10 \\ .13 & .09 & .07 \\ .10 & .15 & .09 \end{bmatrix}, \quad \mathcal{Q}^{(0)} = \begin{bmatrix} .20 & .12 & .13 \\ .12 & .20 & .15 \\ .13 & .15 & .20 \end{bmatrix}.$$

All simulations are averaged over 100 realizations, and the convergence criteria of the EM algorithm is $\frac{||\Theta^{(i)} - \Theta^{(i-1)}||_2}{||\Theta^{(i-1)}||_2} \leq 0.1$. In the third simulation we finally assume that the model is a constant linear Gaussian HMC (with unknown parameters), estimate its parameters via the EM algorithm and restore x thanks to a Kalman filter. As expected, the supervised TMC restoration algorithm outperforms the unsupervised one, which itself outperforms the unsupervised Kalman filter for HMC.



5. REFERENCES

[1] Y. Ephraim and N. Merhav, "Hidden Markov processes," *IEEE Transactions on Information Theory*, vol. 48, no. 6, pp. 1518–69, June 2002.

[2] W. Pieczynski, "Pairwise Markov chains," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 634–39, May 2003.

[3] W. Pieczynski, C. Hulard, and T. Veit, "Triplet Markov chains in hidden signal restoration," in *SPIE International Symposium on Remote Sensing*, Crete, Grece, September 22-27 2002.

[4] W. Pieczynski and F. Desbouvries, "Kalman filtering using pairwise Gaussian models," in *Proceedings of the ICASSP*, Hong-Kong, 2003.

[5] B. Ait-El-Fquih and F. Desbouvries, "Kalman filtering in triplet Markov chains," *accepted for publication, IEEE Tr. Sig. Proc.*, September 2005.

[6] B. Ait-El-Fquih and F. Desbouvries, "Bayesian smoothing algorithms in pairwise and triplet Markov chains," in *Proceedings of the 2005 IEEE Workshop on Statistical Signal Processing (SSP 05)*, Bordeaux, France, July 17-20 2005.

[7] W. Pieczynski and F. Desbouvries, "On Triplet Markov chains," in *Proceedings of the International Symposium on Applied Stochastic Models and Data Analysis (ASMDA 2005)*, Brest, France, May 17-20 2005.

[8] S. Derrode and W. Pieczynski, "Signal and image segmentation using pairwise Markov chains," *IEEE Transactions on Signal Processing*, vol. 52, no. 9, pp. 2477–89, 2004.

[9] P. Lanchantin and W. Pieczynski, "Unsupervised non stationary image segmentation using triplet Markov chains," in *Advanced Concepts for Intelligent Vision Systems (ACVIS 04)*, Brussels, Belgium, August 31 - September 3 2004.

[10] A. P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society (B)*, vol. 39, no. 1, pp. 1–38, 1977.

[11] G. J. McLachlan and T. Khrishnan, *The EM algorithm and extensions*, Wiley series in probability and statistics. Wiley Interscience, 1996.

[12] M. A. Tanner, *Tools for Statistical Inference: Methods for the exploration of posterior distributions and likelihood functions*, Springer-Verlag, New-York, 1996.

[13] R.H. Shumway and D.S. Stoffer, "An approach to time series smoothing and forecasting using the EM algorithm," *J. Time Series Anal.*, 1982.

[14] F. A. Graybill, *Matrices with applications in statistics*, Wadworth, Belmont, second edition, 1983.

[15] J. R. Magnus and H. Neudecker, *Matrix differential calculus with applications in statistics and econometrics*, Wiley, Chichester, GB, revised edition, 1999.