BAYESIAN SMOOTHING ALGORITHMS IN PARTIALLY OBSERVED MARKOV CHAINS

Boujemaa Ait-el-Fquih and François Desbouvries

Institut National des Télécommunications, Evry, France

Abstract. Let $\mathbf{x} = \{\mathbf{x}_n\}_{n\in\mathbb{N}}$ be a hidden process, $\mathbf{y} = \{\mathbf{y}_n\}_{n\in\mathbb{N}}$ an observed process and $\mathbf{r} = \{\mathbf{r}_n\}_{n\in\mathbb{N}}$ some auxiliary process. We assume that $\mathbf{t} = \{\mathbf{t}_n\}_{n\in\mathbb{N}}$ with $\mathbf{t}_n = (\mathbf{x}_n, \mathbf{r}_n, \mathbf{y}_{n-1})$ is a (Triplet) Markov Chain (TMC). TMC are more general than Hidden Markov Chains (HMC) and yet enable the development of efficient restoration and parameter estimation algorithms. This paper is devoted to Bayesian smoothing algorithms for TMC. We first propose twelve algorithms for general TMC. In the Gaussian case, these smoothers reduce to a set of algorithms which include, among other solutions, extensions to TMC of classical Kalman-like smoothing algorithms (originally designed for HMC) such as the RTS algorithms, the Two-Filter algorithms or the Bryson and Frazier algorithm.

Keywords: Hidden Markov Models, Gauss-Markov Chains, Bayesian restoration
PACS: 02.50.-r, 02.50.Ga

INTRODUCTION

An important problem consists in estimating an unobservable process $\mathbf{x} = \{\mathbf{x}_n\}_{n\in\mathbb{N}}$ from an observed one $\mathbf{y} = \{\mathbf{y}_n\}_{n\in\mathbb{N}}$. 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 tutorials [1] [2]). In an HMC $\mathbf{x}$ is first assumed to be a Markov chain (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 $\mathbf{y}$.

On the other hand, Pairwise [3] (PMC) and Triplet [4] MC (TMC) have been introduced recently. A PMC is a model in which the pair $(\mathbf{x}, \mathbf{y})$ is assumed to be an MC. So 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 $\mathbf{x}$. 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. On the other hand, one can extend from HMC to PMC some efficient Bayesian restoration algorithms. In particular, in the linear Gaussian case, the extension to PMC of the Kalman filter has been considered in [5].

The PMC model can be further generalized to the TMC model [4] 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 an auxiliary process $\mathbf{r} = \{\mathbf{r}_n\}_{n\in\mathbb{N}}$. The triplet $\mathbf{t} = (\mathbf{x}, \mathbf{r}, \mathbf{y})$ is called a TMC if $(\mathbf{x}, \mathbf{r}, \mathbf{y})$ is a (vector) MC. 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. TMC indeed include some classical generalizations of HMC [6] [5]. For instance, Hidden semi-Markov Chains [7] [8] [9] are particular TMC with $\mathbf{x}$ and r discrete and in which r_n

represents the time during which $\mathbf{x}_n$ remains in the same state; Jump-Markov state space systems are particular TMC with $\mathbf{x}$ continuous and r discrete; state-space systems with colored process and/or measurement noise(s) are particular TMC with continuous hidden and auxiliary processes.

- 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 [4] or linear Gaussian [5] cases).

The wider generality of PMC w.r.t. HMC and of TMC w.r.t. PMC can also be seen through the expression of the conditional law of $\mathbf{y}$ given $\mathbf{x}$. In an HMC it is classically assumed that $p(\mathbf{y}_{0:n}|\mathbf{x}_{0:n}) = p(\mathbf{y}_0|\mathbf{x}_0)\cdots p(\mathbf{y}_n|\mathbf{x}_n)$, which is very simple, and indeed too simple in such applications as speech recognition [10] [11]; in a PMC $p(\mathbf{y}_{0:n}|\mathbf{x}_{0:n})$ is an MC, which is much richer; and in a TMC $p(\mathbf{y}_{0:n}|\mathbf{x}_{0:n})$ is the marginal distribution of the MC $p(\mathbf{y}_{0:n},\mathbf{r}_{0:n}|\mathbf{x}_{0:n})$, which is still much richer than an MC. PMC and TMC models are thus expected to better fit the need for a better modeling of the noise distribution.

Let us now turn to the contribution of this paper. We first propose twelve smoothing algorithms for general continuous TMC. These algorithms are derived from Markovian properties of $\mathbf{t}$ only, considered as an MC both in the forward and the backward direction. They can be classified into three classes : four forward filtering backward smoothing algorithms, four backward filtering forward smoothing algorithms and four non-recursive algorithms. We emphasize on the role played by four probability density functions (pdf) (α_n, β_n, γ_n and δ_n) as well as by the (forward and/or backward) TMC pdf of the MC $\mathbf{t}$.

We next address the particular case of Gaussian TMC. The general algorithms then reduce to twelve specific algorithms (plus variations thereof). This section extends the family of existing fixed interval smoothing algorithms in two directions. On the one hand, our algorithms are directly proposed for the TMC framework and can be particularized (if necessary) to the HMC case; on the other hand, even in the HMC case our set of algorithms encompasses some classical solutions (RTS algorithm in first [12] and second [13] form, Two-Filter algorithms [14] [15], Bryson and Frazier algorithm [16] ...) but it also contains some original algorithms.

TMC FIXED-INTERVAL SMOOTHING ALGORITHMS

Let $\mathbf{x}_n \in \mathbb{R}^{n_\mathbf{x}}$ be the hidden process, $\mathbf{y}_n \in \mathbb{R}^{n_\mathbf{y}}$ the observation and $\mathbf{r}_n \in \mathbb{R}^{n_\mathbf{r}}$ the auxiliary process. Let us set $\mathbf{x}^*_n = (\mathbf{x}_n,\mathbf{r}_n) \in \mathbb{R}^{n_{\mathbf{x}^*}}$ and $\mathbf{t}_n = (\mathbf{x}_n,\mathbf{r}_n,\mathbf{y}_{n-1})$. We assume that $\mathbf{t} = \{\mathbf{t}_n\}_{n\geq 0}$ (with $\mathbf{y}_{-1} = \mathbf{0}$) is an MC. Let $p(\mathbf{x}_{0:n})$ (resp. $p(\mathbf{x}^*_n|\mathbf{y}_{0:n})$), say, denote the pdf (w.r.t. Lebesgue measure) of $\mathbf{x}_{0:n}$ (resp. of $\mathbf{x}^*_n$ given $\mathbf{y}_{0:n}$); other pdfs of interest are defined similarly. The aim of this section is to propose general fixed-interval Bayesian smoothing algorithms for TMC, i.e. we want to compute the smoothing pdf $p(\mathbf{x}_n|\mathbf{y}_{0:N})$ for all n, $0 \leq n \leq N$. In the following we indeed focus on the computation of $p(\mathbf{x}^*_n|\mathbf{y}_{0:N})$; the pdf $p(\mathbf{x}_n|\mathbf{y}_{0:N})$ of interest is obtained by marginalization.

The algorithms we propose can be classified into three families :

1. *Backward recursive algorithms.* These are two-pass algorithms, in which (i) $p(\mathbf{x}_n^*|\mathbf{y}_{0:N})$ is computed from $p(\mathbf{x}_{n+1}^*|\mathbf{y}_{0:N})$ via
$$p(\mathbf{x}_n^*|\mathbf{y}_{0:N}) = \int_{\mathbb{R}^{n_{\mathbf{x}^*}}} p(\mathbf{x}_{n+1}^*|\mathbf{y}_{0:N}) p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*, \mathbf{y}_{0:N}) d\mathbf{x}_{n+1}^* \quad (1)$$
(whence the term "backward recursive algorithm"); and (ii) $p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*, \mathbf{y}_{0:N})$ in (1) is computed in the forward direction;
2. *Forward recursive algorithms.* These are two-pass algorithms, in which (i) $p(\mathbf{x}_{n+1}^*|\mathbf{y}_{0:N})$ is computed from $p(\mathbf{x}_n^*|\mathbf{y}_{0:N})$ via
$$p(\mathbf{x}_{n+1}^*|\mathbf{y}_{0:N}) = \int_{\mathbb{R}^{n_{\mathbf{x}^*}}} p(\mathbf{x}_n^*|\mathbf{y}_{0:N}) p(\mathbf{x}_{n+1}^*|\mathbf{x}_n^*, \mathbf{y}_{0:N}) d\mathbf{x}_n^*, \quad (2)$$
and (ii) $p(\mathbf{x}_{n+1}^*|\mathbf{x}_n^*, \mathbf{y}_{0:N})$ in (2) is computed in the backward direction;
3. *Non-recursive algorithms.* In these algorithms, $p(\mathbf{x}_n^*|\mathbf{y}_{0:N})$ is computed from two pdfs; one of them is computed recursively in the forward direction and the other recursively in the backward direction.

These algorithms can be further sub-classified by taking into account which TMC pdfs are used in the computations. More precisely, since **t** is an MC **t** is also an MC in the backward direction; so we have

$$\begin{aligned} p(\mathbf{x}_{n+1}^*, \mathbf{y}_n|\mathbf{x}_{0:n}^*, \mathbf{y}_{0:n-1}) &= p(\mathbf{x}_{n+1}^*, \mathbf{y}_n|\mathbf{x}_n^*, \mathbf{y}_{n-1}) \\ &= p(\mathbf{x}_{n+1}^*|\mathbf{x}_n^*, \mathbf{y}_n, \mathbf{y}_{n-1}) p(\mathbf{y}_n|\mathbf{x}_n^*, \mathbf{y}_{n-1}), \quad (3) \\ p(\mathbf{x}_n^*, \mathbf{y}_{n-1}|\mathbf{x}_{n+1:N}^*, \mathbf{y}_{n:N}) &= p(\mathbf{x}_n^*, \mathbf{y}_{n-1}|\mathbf{x}_{n+1}^*, \mathbf{y}_n) \\ &= p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*, \mathbf{y}_n, \mathbf{y}_{n-1}) p(\mathbf{y}_{n-1}|\mathbf{x}_{n+1}^*, \mathbf{y}_n). \quad (4) \end{aligned}$$

As we are going to see, each of the three families of algorithms above contains one algorithm which only uses the forward TMC pdfs $p(\mathbf{x}_{n+1}^*|\mathbf{x}_n^*, \mathbf{y}_n, \mathbf{y}_{n-1})$ and $p(\mathbf{y}_n|\mathbf{x}_n^*, \mathbf{y}_{n-1})$ in (3), one algorithm which only uses the backward TMC pdfs $p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*, \mathbf{y}_n, \mathbf{y}_{n-1})$ and $p(\mathbf{y}_{n-1}|\mathbf{x}_{n+1}^*, \mathbf{y}_n)$ in (4), and two algorithms which use both.

Let us give two familiar examples in the case where $\mathbf{x}_n^* = \mathbf{x}_n$ and $(\mathbf{x}, \mathbf{y})$ is a classical HMC, i.e. in the case where there is no auxiliary process **r** and the factorization $p(\mathbf{x}_{0:n}, \mathbf{y}_{0:n}) = p(\mathbf{x}_0) \prod_{i=1}^n p(\mathbf{x}_i|\mathbf{x}_{i-1}) \prod_{i=0}^n p(\mathbf{y}_i|\mathbf{x}_i)$ holds. Then in factorization (3) $p(\mathbf{x}_{n+1}^*|\mathbf{x}_n^*, \mathbf{y}_n, \mathbf{y}_{n-1})$ reduces to $p(\mathbf{x}_{n+1}|\mathbf{x}_n)$ and $p(\mathbf{y}_n|\mathbf{x}_n^*, \mathbf{y}_{n-1})$ reduces to $p(\mathbf{y}_n|\mathbf{x}_n)$. If **x** is discrete, the Forward-Backward algorithm [17] computes $p(\mathbf{x}_n|\mathbf{y}_{0:N})$ as a (normalized) product of two pdfs $\widetilde{\alpha}(\mathbf{x}_n)$ and $\widetilde{\beta}(\mathbf{x}_n)$, in which $\widetilde{\alpha}(\mathbf{x}_n)$ (resp. $\widetilde{\beta}(\mathbf{x}_n)$) is computed recursively in the forward (resp. backward) direction, by recursions which use only the forward HMC pdfs $p(\mathbf{x}_{n+1}|\mathbf{x}_n)$ and $p(\mathbf{y}_n|\mathbf{x}_n)$; the Forward-Backward algorithm thus belongs to the third family of algorithms (first subclass). On the other hand, in the Gaussian case the RTS algorithm [12] belongs to the first family (first subclass).

Let us now get more precisely into the contents of this section. As we shall see, each one of the algorithms (9)-(20) makes use of one (or two) out of the four pdfs $\alpha_n \stackrel{\text{def}}{=}$

$p(\mathbf{x}_n^*|\mathbf{y}_{0:n-1})$, $\beta_n \stackrel{\text{def}}{=} p(\mathbf{y}_{n:N}|\mathbf{t}_n)$, $\gamma_n \stackrel{\text{def}}{=} p(\mathbf{x}_n^*|\mathbf{y}_{n-1:N})$ and $\delta_n \stackrel{\text{def}}{=} p(\mathbf{y}_{0:n-2}|\mathbf{t}_n)$. These pdfs, in turn, can be computed recursively (in the forward direction for α_n and δ_n, in the backward direction for β_n and γ_n) from the (either forward or backward) TMC pdfs; so for sake of clarity let us first gather these recursions in equations (5) to (8).

Recursive algorithms for α_n, β_n, γ_n and δ_n

Let us classify the recursive algorithms for α_n, β_n, γ_n and δ_n according to which TMC pdfs are used in the computations : $p(\mathbf{x}_{n+1}^*|\mathbf{x}_n^*,\mathbf{y}_n,\mathbf{y}_{n-1})$ and $p(\mathbf{y}_n|\mathbf{x}_n^*,\mathbf{y}_{n-1})$ for the "forward" recursive algorithms (see factorisation (3)), and $p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*,\mathbf{y}_n,\mathbf{y}_{n-1})$ and $p(\mathbf{y}_{n-1}|\mathbf{x}_{n+1}^*,\mathbf{y}_n)$ for the "backward" recursive algorithms (see factorisation (4)).

The algorithm described in Proposition 1 propagates α_n in the forward direction and β_n in the backward direction.

Proposition 1 *Assume that we are given the forward TMC pdfs (see factorisation (3)) of the MC* $\mathbf{t}$*. Then the one-step prediction pdf* $\alpha_n = p(\mathbf{x}_n^*|\mathbf{y}_{0:n-1})$ *and the filtering pdf* $\widetilde{\alpha}_n = p(\mathbf{x}_n^*|\mathbf{y}_{0:n})$ *can be propagated from* $n = 0$ *to* N *(with* $\alpha_0 = p(\mathbf{x}_0^*)$*) as*

$$\left\{ \begin{array}{lcl} p(\mathbf{x}_n^*|\mathbf{y}_{0:n}) & = & \frac{p(\mathbf{y}_n|\mathbf{t}_n)p(\mathbf{x}_n^*|\mathbf{y}_{0:n-1})}{\int_{\mathbb{R}^{n_{\mathbf{x}^*}}} p(\mathbf{y}_n|\mathbf{t}_n)p(\mathbf{x}_n^*|\mathbf{y}_{0:n-1})d\mathbf{x}_n^*} \\ p(\mathbf{x}_{n+1}^*|\mathbf{y}_{0:n}) & = & \int_{\mathbb{R}^{n_{\mathbf{x}^*}}} p(\mathbf{x}_{n+1}^*|\mathbf{t}_n,\mathbf{y}_n)p(\mathbf{x}_n^*|\mathbf{y}_{0:n})d\mathbf{x}_n^* \end{array} \right. ; \tag{5}$$

on the other hand, the likelihood functions $\beta_n = p(\mathbf{y}_{n:N}|\mathbf{t}_n)$ *and* $\widetilde{\beta}_n = p(\mathbf{y}_{n+1:N}|\mathbf{t}_n,\mathbf{y}_n)$ *can be computed from* $n = N$ *to* $n = 0$ *(with* $\beta_{N+1} = 1$*) as*

$$\left\{ \begin{array}{lcl} \widetilde{\beta}_n & = & \int_{\mathbb{R}^{\mathbf{x}_n^*}} p(\mathbf{x}_{n+1}^*|\mathbf{t}_n,\mathbf{y}_n) \times \beta_{n+1}\, d\mathbf{x}_{n+1}^* \\ \beta_n & = & p(\mathbf{y}_n|\mathbf{t}_n) \times \widetilde{\beta}_n \end{array} \right. . \tag{6}$$

The algorithm described in Proposition 2 propagates δ_n in the forward and γ_n in the backward direction. These recursions now only make use of the backward TMC pdfs $p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*,\mathbf{y}_n,\mathbf{y}_{n-1})$ and $p(\mathbf{y}_{n-1}|\mathbf{x}_{n+1}^*,\mathbf{y}_n)$.

Proposition 2 *Assume that we are given the backward TMC pdfs* $p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*,\mathbf{y}_n,\mathbf{y}_{n-1})$ *and* $p(\mathbf{y}_{n-1}|\mathbf{x}_{n+1}^*,\mathbf{y}_n)$ *(see factorisation (4)) of the MC* $\mathbf{t}$*. Then the likelihood functions* $\delta_n = p(\mathbf{y}_{0:n-2}|\mathbf{t}_n)$ *and* $\widetilde{\delta}_{n+1} = p(\mathbf{y}_{0:n-2}|\mathbf{t}_{n+1},\mathbf{y}_{n-1})$ *can be computed from* $n = 1$ *to* N *(with* $\delta_1 = 1$*) as*

$$\left\{ \begin{array}{lcl} \widetilde{\delta}_{n+1} & = & \int_{\mathbb{R}^{n_{\mathbf{x}^*}}} p(\mathbf{x}_n^*|\mathbf{t}_{n+1},\mathbf{y}_{n-1}) \times \delta_n\, d\mathbf{x}_n^* \\ \delta_{n+1} & = & p(\mathbf{y}_{n-1}|\mathbf{t}_{n+1}) \times \widetilde{\delta}_{n+1} \end{array} \right. ; \tag{7}$$

on the other hand, $\gamma_n = p(\mathbf{x}_n^*|\mathbf{y}_{n-1:N})$ *and* $\widetilde{\gamma}_{n+1} = p(\mathbf{x}_{n+1}^*|\mathbf{y}_{n-1:N})$ *can be computed from* $n = N$ *to* $n = 1$ *(with* $p(\mathbf{x}_{N+1}^*|\mathbf{y}_N) = \frac{p(\mathbf{t}_{N+1})}{p(\mathbf{y}_N)}$*) as*

$$\left\{ \begin{array}{lcl} p(\mathbf{x}_{n+1}^*|\mathbf{y}_{n-1:N}) & = & \frac{p(\mathbf{y}_{n-1}|\mathbf{t}_{n+1})p(\mathbf{x}_{n+1}^*|\mathbf{y}_{n:N})}{\int_{\mathbb{R}^{n_{\mathbf{x}^*}}} p(\mathbf{y}_{n-1}|\mathbf{t}_{n+1})p(\mathbf{x}_{n+1}^*|\mathbf{y}_{n:N})d\mathbf{x}_{n+1}^*} \\ p(\mathbf{x}_n^*|\mathbf{y}_{n-1:N}) & = & \int_{\mathbb{R}^{n_{\mathbf{x}^*}}} p(\mathbf{x}_n^*|\mathbf{t}_{n+1},\mathbf{y}_{n-1})p(\mathbf{x}_{n+1}^*|\mathbf{y}_{n-1:N})d\mathbf{x}_{n+1}^* \end{array} \right. . \tag{8}$$

Backward recursive computation of the smoothing pdf

The aim of this section is to compute the backward conditional TMC pdf $p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*,\mathbf{y}_{0:N})$ in equation (1). Since $\mathbf{t}$ is an MC, $\mathbf{y}_{n+1:N}$ and $\mathbf{x}_n^*$ are independent conditionally on $(\mathbf{x}_{n+1}^*,\mathbf{y}_{0:n})$, so $p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*,\mathbf{y}_{0:N}) = p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*,\mathbf{y}_{0:n})$. As we now see, $p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*,\mathbf{y}_{0:n})$ can be computed by combining appropriately $\widetilde{\alpha}_n$, α_n or δ_n and the forward (or backward) TMC pdfs, which leads to four different algorithms. The algorithm (9) (resp. (10)) only uses forward (resp. backward) TMC pdfs, and the algorithms (11) and (12) use both.

Proposition 3 *Assume that we are given the forward and/or backward TMC pdfs (see factorisations (3) and (4)) of the MC* $\mathbf{t}$*. Then* $\widetilde{\alpha}_n$ *and* α_{n+1} *(resp.* δ_n*) can be computed in the forward direction by (5) (resp. (7)), and next* $p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*,\mathbf{y}_{0:n})$ *by*

$$p(\mathbf{x}_n^*|\mathbf{x}_{n+1}^*,\mathbf{y}_{0:n}) = \frac{p(\mathbf{x}_{n+1}^*|\mathbf{t}_n,\mathbf{y}_n)\widetilde{\alpha}_n}{\int_{\mathbb{R}^{n_{\mathbf{x}^*}}} p(\mathbf{x}_{n+1}^*|\mathbf{t}_n,\mathbf{y}_n)\widetilde{\alpha}_n d\mathbf{x}_n^*} \tag{9}$$

$$= \begin{cases} p(\mathbf{x}_0^*|\mathbf{t}_1) & \text{if } n=0 \\ \frac{p(\mathbf{x}_n^*|\mathbf{t}_{n+1},\mathbf{y}_{n-1})\delta_n}{\int_{\mathbb{R}^{n_{\mathbf{x}^*}}} p(\mathbf{x}_n^*|\mathbf{t}_{n+1},\mathbf{y}_{n-1})\delta_n d\mathbf{x}_n^*} & \text{if } n \geq 1 \end{cases} \tag{10}$$

$$= \frac{\frac{p(\mathbf{x}_n^*|\mathbf{t}_{n+1},\mathbf{y}_{n-1})\alpha_n}{p(\mathbf{x}_n^*|\mathbf{y}_{n-1})}}{\int_{\mathbb{R}^{n_{\mathbf{x}^*}}} \frac{p(\mathbf{x}_n^*|\mathbf{t}_{n+1},\mathbf{y}_{n-1})\alpha_n}{p(\mathbf{x}_n^*|\mathbf{y}_{n-1})} d\mathbf{x}_n^*} \tag{11}$$

$$= \frac{p(\mathbf{t}_{n+1}|\mathbf{t}_n)\delta_n p(\mathbf{x}_n^*|\mathbf{y}_{n-1})}{\int_{\mathbb{R}^{n_{\mathbf{x}^*}}} p(\mathbf{t}_{n+1}|\mathbf{t}_n)\delta_n p(\mathbf{x}_n^*|\mathbf{y}_{n-1}) d\mathbf{x}_n^*}. \tag{12}$$

Finally $p(\mathbf{x}_n^*|\mathbf{y}_{0:N})$ *can be computed in reverse-time (from* $n = N$ *to* $n = 0$*) by (1).*

Forward recursive computation of the smoothing pdf

This section is parallel to the previous one. Our aim here is to compute $p(\mathbf{x}_{n+1}^*|\mathbf{x}_n^*,\mathbf{y}_{0:N})$ in (2). Since $\mathbf{t}$ is an MC, $\mathbf{y}_{0:n-2}$ and $\mathbf{x}_{n+1}^*$ are independent conditionally on $(\mathbf{x}_n^*,\mathbf{y}_{n-1:N})$, so $p(\mathbf{x}_{n+1}^*|\mathbf{x}_n^*,\mathbf{y}_{0:N}) = p(\mathbf{x}_{n+1}^*|\mathbf{x}_n^*,\mathbf{y}_{n-1:N})$. As we now see, $p(\mathbf{x}_{n+1}^*|\mathbf{x}_n^*,\mathbf{y}_{n-1:N})$ can be computed by combining appropriately β_n, γ_n or $\widetilde{\gamma}_n$ or and the forward (or backward) TMC pdfs, which leads to four different algorithms. The algorithm (13) (resp. (14)) only uses forward (resp. backward) TMC pdfs, and the algorithms (15) and (16) use both.

Proposition 4 *Assume that we are given the forward and/or backward TMC pdfs (see factorisations (3) and (4)) of the MC* $\mathbf{t}$*. Then* β_n *(resp.* γ_n *and* $\widetilde{\gamma}_n$*) can be computed in the backward direction by (6) (resp. (8)), and next* $p(\mathbf{x}_{n+1}^*|\mathbf{x}_n^*,\mathbf{y}_{n-1:N})$ *by*

$$p(\mathbf{x}_{n+1}^*|\mathbf{x}_n^*,\mathbf{y}_{n-1:N}) = \frac{p(\mathbf{x}_{n+1}^*|\mathbf{t}_n,\mathbf{y}_n)\beta_{n+1}}{\int_{\mathbb{R}^{n_{\mathbf{x}^*}}} p(\mathbf{x}_{n+1}^*|\mathbf{t}_n,\mathbf{y}_n)\beta_{n+1} d\mathbf{x}_{n+1}^*} \tag{13}$$

$$= \frac{p(\mathbf{x}_n^*|\mathbf{t}_{n+1},\mathbf{y}_{n-1})\widetilde{\gamma}_{n+1}}{\int_{\mathbb{R}^{n_{\mathbf{x}^*}}} p(\mathbf{x}_n^*|\mathbf{t}_{n+1},\mathbf{y}_{n-1})\widetilde{\gamma}_{n+1} d\mathbf{x}_{n+1}^*} \tag{14}$$

$$= \frac{\frac{p(\mathbf{x}^*_{n+1}|\mathbf{t}_n,\mathbf{y}_n)\gamma_{n+1}}{p(\mathbf{x}^*_{n+1}|\mathbf{y}_n)}}{\int_{\mathbb{R}^{n_{\mathbf{x}^*}}} \frac{p(\mathbf{x}^*_{n+1}|\mathbf{t}_n,\mathbf{y}_n)\gamma_{n+1}}{p(\mathbf{x}^*_{n+1}|\mathbf{y}_n)} d\mathbf{x}^*_{n+1}} \tag{15}$$

$$= \frac{p(\mathbf{t}_n|\mathbf{t}_{n+1})\beta_{n+1}p(\mathbf{x}^*_{n+1}|\mathbf{y}_n)}{\int_{\mathbb{R}^{n_{\mathbf{x}^*}}} p(\mathbf{t}_n|\mathbf{t}_{n+1})\beta_{n+1}p(\mathbf{x}^*_{n+1}|\mathbf{y}_n)d\mathbf{x}^*_{n+1}}. \tag{16}$$

Finally $p(\mathbf{x}^*_n|\mathbf{y}_{0:N})$ can be computed in the forward direction by (2).

Non recursive computation of the smoothing pdf

Let us now see that $p(\mathbf{x}^*_n|\mathbf{y}_{0:N})$ can be essentially computed as a (normalized) product of α_n (or δ_n) and β_n (or γ_n), which leads to four algorithms. The algorithm (17) (resp. (18)) only uses forward (resp. backward) TMC pdfs, and (19) and (20) use both.

Proposition 5 *Assume that we are given the forward and/or backward TMC pdfs (see factorisations (3) and (4)) of the MC* $\mathbf{t}$*. Then the smoothing pdf can be computed as*

$$p(\mathbf{x}^*_n|\mathbf{y}_{0:N}) = \frac{\alpha_n \times \beta_n}{\int_{\mathbb{R}^{n_{\mathbf{x}^*}}} \alpha_n \times \beta_n d\mathbf{x}^*_n} \tag{17}$$

$$= \frac{\gamma_n \times \delta_n}{\int_{\mathbb{R}^{n_{\mathbf{x}^*}}} \gamma_n \times \delta_n d\mathbf{x}^*_n} \tag{18}$$

$$= \frac{\frac{\alpha_n \times \gamma_n}{p(\mathbf{x}^*_n|\mathbf{y}_{n-1})}}{\int_{\mathbb{R}^{n_{\mathbf{x}^*}}} \frac{\alpha_n \times \gamma_n}{p(\mathbf{x}^*_n|\mathbf{y}_{n-1})} d\mathbf{x}^*_n} \tag{19}$$

$$= \frac{\delta_n \times \beta_n \times p(\mathbf{x}^*_n|\mathbf{y}_{n-1})}{\int_{\mathbb{R}^{n_{\mathbf{x}^*}}} \delta_n \times \beta_n \times p(\mathbf{x}^*_n|\mathbf{y}_{n-1}) d\mathbf{x}^*_n}, \tag{20}$$

in which α_n (resp. δ_n) is computed in the forward direction by (5) (resp. (7)), and β_n (resp. γ_n) is computed in the backward direction by (6) (resp. (8)).

THE GAUSSIAN CASE

The aim of this section is to derive Kalman-like smoothing algorithms for Gaussian TMC. From now on we thus assume that

$$\underbrace{\begin{bmatrix} \mathbf{x}^*_{n+1} \\ \mathbf{y}_n \end{bmatrix}}_{\mathbf{t}_{n+1}} = \begin{bmatrix} \mathscr{F}_n^{\mathbf{x}^*,\mathbf{x}^*} & \mathscr{F}_n^{\mathbf{x}^*,\mathbf{y}} \\ \mathscr{F}_n^{\mathbf{y},\mathbf{x}^*} & \mathscr{F}_n^{\mathbf{y},\mathbf{y}} \end{bmatrix} \begin{bmatrix} \mathbf{x}^*_n \\ \mathbf{y}_{n-1} \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{w}_n^{\mathbf{x}^*} \\ \mathbf{w}_n^{\mathbf{y}} \end{bmatrix}}_{\mathbf{w}_n}, \tag{21}$$

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

$$\mathbf{w}_n \sim \mathscr{N}(\mathbf{0}, \underbrace{\begin{bmatrix} \mathscr{Q}_n^{\mathbf{x}^*,\mathbf{x}^*} & \mathscr{Q}_n^{\mathbf{x}^*,\mathbf{y}} \\ \mathscr{Q}_n^{\mathbf{y},\mathbf{x}^*} & \mathscr{Q}_n^{\mathbf{y},\mathbf{y}} \end{bmatrix}}_{\mathscr{Q}_n}). \tag{22}$$

We also assume that $\mathbf{P}_0^{\mathbf{x}^*,\mathbf{x}^*}$ and $\mathscr{Q}_n$ are positive definite. If $\mathbf{x}_n^* = \mathbf{x}_n$, $\mathscr{F}_n^{\mathbf{x}^*,\mathbf{y}} = \mathbf{0}$, $\mathscr{F}_n^{\mathbf{y},\mathbf{y}} = \mathbf{0}$ and $\mathscr{Q}_n^{\mathbf{x}^*,\mathbf{y}} = \mathbf{0}$, model (21)-(22) reduces to the classical state-space system.

From (21) and (22), $\mathbf{t}$ is a vector Gauss-Markov process. So all the pdfs in the last section are Gaussian, and in particular $p(\mathbf{x}_n^*|\mathbf{y}_{0:N}) \sim \mathscr{N}(\widehat{\mathbf{x}}_{n|0:N}^*, \mathbf{P}_{n|0:N}^{\mathbf{x}^*,\mathbf{x}^*})$. Computing $p(\mathbf{x}_n^*|\mathbf{y}_{0:N})$ amounts to computing $\widehat{\mathbf{x}}_{n|0:N}^*$ and $\mathbf{P}_{n|0:N}^{\mathbf{x}^*,\mathbf{x}^*}$, and indeed the general algorithms of Propositions 3 to 5 (which compute $p(\mathbf{x}_n^*|\mathbf{y}_{0:N})$ from α_n (or δ_n) and/or γ_n (or β_n)), reduce to equations which compute $\arg\max_{\mathbf{x}_n^*} p(\mathbf{x}_n^*|\mathbf{y}_{0:N})$ (i.e., $\widehat{\mathbf{x}}_{n|0:N}^*$), and the associated covariance matrix, from $\arg\max_{\mathbf{x}_n^*} \alpha_n = \widehat{\mathbf{x}}_{n|0:n-1}^*$ (or $\arg\max_{\mathbf{x}_n^*} \delta_n$) and/or $\arg\max_{\mathbf{x}_n^*} \gamma_n = \widehat{\mathbf{x}}_{n|n-1:N}^*$ (or $\arg\max_{\mathbf{x}_n^*} \beta_n$), as well as the associated covariance matrice(s). It is not possible here to write down these twelve Gaussian algorithms (plus variations thereof) explicitely. Let us just mention that they are most of the time extensions to the TMC context of algorithms already proposed in the classical state-space smoothing literature. More precisely:

- *Recursive algorithms for α_n, β_n, γ_n and δ_n.*
  - Equation (5) reduces to an algorithm which, in filter form [5] (resp. in information form) is an extension to TMC of the Kalman filter in filter form [18] (resp. in information form [19] [16]);
  - (6) reduces to an algorithm which propagates $\arg\max_{\mathbf{x}_n^*} \beta_n$. After some manipulations, one can easily show that it generalizes to TMC the backward algorithm used in the two-filter smoother by Mayne [14] [16, eqs. (10.4.14)-(10.4.15)];
  - (7) reduces to an algorithm which propagates $\arg\max_{\mathbf{x}_n^*} \delta_n$. After some manipulations, one can easily show that it has a counterpart in the HMC case, introduced in the context of complementary models by Weinert (see [20, §3.2]);
  - equation (8) reduces to filter or information forms algorithms; in particular, the filter form algorithm has an HMC counterpart [16, §9.8]).
- *Backward recursive computation of the smoothing pdf.*
  - Equations (1) and (9) reduce to an algorithm [21, Prop. 3] which is an extension to TMC of the RTS algorithm [12];
  - equations (1) and (10) reduce to an algorithm which extends to TMC an algorithm introduced by Weinert [20, p. 40] (in the case $\mathscr{F}_n^{\mathbf{x}^*,\mathbf{x}^*}$ invertible);
  - finally, equations ((1) and (11)), and ((1) and (12)), reduce to algorithms which, to our best knowledge, have no counterparts in the HMC case.
- *Forward recursive computation of the smoothing pdf.*
  - Equations (2) and (13) reduce to an algorithm which extends to TMC an algorithm introduced in the concept of complementary models by Desai *et al.* [13] [20, p. 35];
  - equations (2) and (14) reduce to an algorithm which extends to TMC an algorithm partially[1] found in Kailath *et al.* [16, pp. 401, Exs. 10.12 & 10.14];

---

[1] In fact, only the mean of the smoothing pdf is given explicitely.

  - equations ((2) and (15)), and ((2) and (16)), reduce to algorithms which, to our best knowledge, have no counterparts in the HMC case.
- *Non recursive computation of the smoothing pdf.*
  - Equation (17) reduces to an algorithm which extends to TMC the two-filter algorithm by Mayne [14] (see also Fraser and Potter [15]);
  - equation (18) reduces to an algorithm which, to the best of our knowledge, has no counterpart in the HMC case;
  - equation (19) reduces to an algorithm [21, Prop. 4] which extends to TMC the General two-filter algorithm [16, Thm. 10.4.1];
  - finally, one can show after some computations that equation (20) reduces to an algorithm which extends to TMC an algorithm introduced by Weinert [20, §3.3] in the context of complementary models.

REFERENCES

1. Y. Ephraim, and N. Merhav, *IEEE Transactions on Information Theory* **48**, 1518–69 (2002).
2. O. Cappé, E. Moulines, and T. Rydén, *Inference in Hidden Markov Models*, Springer-Verlag, 2005.
3. W. Pieczynski, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **25**, 634–39 (2003).
4. W. Pieczynski, C. Hulard, and T. Veit, "Triplet Markov Chains in hidden signal restoration," in *SPIE International Symposium on Remote Sensing*, Crete, Grece, 2002.
5. B. Ait-El-Fquih, and F. Desbouvries, *IEEE Transactions on Signal Processing* **54**, 2957–63 (2006).
6. 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, 2005.
7. S.-Z. Yu, and H. Kobayashi, *Signal Processing* **83**, 235–250 (2003).
8. M. D. Moore, and M. I. Savic, *Digital Signal Processing* **14**, 37–53 (2004).
9. Y. Guédon, *Computational Statistics and Data Analysis* **49**, 663–688 (2005).
10. C. J. Wellekens, "Explicit time correlation in hidden Markov models for speech recognition," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP 87)*, 1987, vol. 12, pp. 384–86.
11. M. Ostendorf, V. Digalakis, and O. A. Kimball, *IEEE Transactions on Speech and Audio Processing* **4**, 360–378 (1996).
12. H. E. Rauch, F. Tung, and C. T. Striebel, *AIAA J.* **3**, 1445–1450 (1965).
13. U. Desai, H. Weinert, and G. Yusypchuk, *IEEE Transactions on Automatic Control* **28**, 536–539 (1983).
14. D. Q. Mayne, *Automatica* **4**, 73–92 (1966).
15. D. C. Fraser, and J. E. Potter, *IEEE Transactions on Automatic Control* **7**, 387–90 (1969).
16. T. Kailath, A. H. Sayed, and B. Hassibi, *Linear estimation*, Prentice Hall Information and System Sciences Series, Prentice Hall, Upper Saddle River, New Jersey, 2000.
17. L. E. Baum, and T. Petrie, *Ann. Math. Stat.* **37**, 1554–63 (1966).
18. R. E. Kalman, *J. Basic Eng., Trans. ASME, Series D* **82**, 35–45 (1960).
19. B. D. O. Anderson, and J. B. Moore, *Optimal Filtering*, Prentice Hall, Englewood Cliffs, New Jersey, 1979.
20. H. L. Weinert, *Fixed interval smoothing for state-space models*, The Kluwer international series in engineering and computer science, Kluwer Academic Publishers, Boston, 2001.
21. 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, 2005.