# FAST FILTER IN NON-LINEAR SYSTEMS WITH APPLICATION TO STOCHASTIC VOLATILITY MODEL

*S. Derrode*

École Centrale Lyon,
LIRIS, CNRS UMR 5205
69134 Ecully, France.

*W. Pieczynski*

Télécom SudParis, Institut Mines-Télécom
CITI Department, CNRS UMR 5157
91011 Evry cedex, France.

## ABSTRACT

We consider the problem of optimal statistical filtering in non-linear and non-Gaussian systems. The novelty consists of approximating the non-linear system by a recent switching system, in which exact fast optimal filtering is workable. The new method is applied to filter stochastic volatility model and some experiments show its efficiency.

***Index Terms***— Non-linear systems, Stochastic volatility model, Optimal statistical filter, Conditionally Gaussian linear state-space model, Conditionally Markov switching hidden linear model, Filtering in switching systems.

## 1. INTRODUCTION

Let us consider two random sequences $\mathbf{X}_1^N = (\mathbf{X}_1, \ldots, \mathbf{X}_N)$ and $\mathbf{Y}_1^N = (\mathbf{Y}_1, \ldots, \mathbf{Y}_N)$, taking their values in $\mathbb{R}^m$ and $\mathbb{R}^q$ respectively. $\mathbf{X}_1^N$ is hidden, while $\mathbf{Y}_1^N$ is observed. The "optimal filter" problem we deal with in this paper consists of the sequential search of $\mathbf{X}_1^N$ from $\mathbf{Y}_1^N$. More precisely, with usual notations for conditional expectations and variances, we search $\mathrm{E}\left[\mathbf{X}_{n+1} \left| \mathbf{y}_1^{n+1} \right.\right]$ and $\mathrm{E}\left[\mathbf{X}_{n+1}\mathbf{X}_{n+1}^T \left| \mathbf{y}_1^{n+1} \right.\right]$ from $\mathrm{E}\left[\mathbf{X}_n \left| \mathbf{y}_1^n \right.\right]$, $\mathrm{E}\left[\mathbf{X}_n\mathbf{X}_n^T \left| \mathbf{y}_1^n \right.\right]$ and $\mathbf{y}_{n+1}$.

The modelling of the distribution $p\left(\mathbf{x}_1^N, \mathbf{y}_1^N\right)$ of the couple $(\mathbf{X}_1^N, \mathbf{Y}_1^N)$ we deal with in this paper is the classical hidden Markov chain, which can be neither Gaussian nor linear. Thus $p\left(\mathbf{x}_1^N, \mathbf{y}_1^N\right)$ is defined by $p\left(\mathbf{x}_1, \mathbf{y}_1\right)$ and recursions:

$$\mathbf{X}_{n+1} = F(\mathbf{X}_n, \mathbf{U}_{n+1}); \quad (1)$$
$$\mathbf{Y}_{n+1} = G(\mathbf{X}_{n+1}, \mathbf{V}_{n+1}), \quad (2)$$

where $\mathbf{U}_1$, $\mathbf{V}_1$, $\ldots \mathbf{U}_N$, $\mathbf{V}_N$ are appropriate independent variables. We will consider the stationary case, which means that the distributions $p\left(\mathbf{x}_n, \mathbf{x}_{n+1}\right)$, $p\left(\mathbf{y}_n \left| \mathbf{x}_n \right.\right)$ resulting from (1), (2) do not depend on $n$. Thus the whole distribution $p\left(\mathbf{x}_1^N, \mathbf{y}_1^N\right)$ is defined by the distribution $p\left(\mathbf{x}_1, \mathbf{y}_1, \mathbf{x}_2, \mathbf{y}_2\right) = p\left(\mathbf{x}_1, \mathbf{x}_2\right) p\left(\mathbf{y}_1 \left| \mathbf{x}_1 \right.\right) p\left(\mathbf{y}_2 \left| \mathbf{x}_2 \right.\right)$, which gives $p\left(\mathbf{x}_1, \mathbf{y}_1\right)$ (equal to $p\left(\mathbf{x}_n, \mathbf{y}_n\right)$ for any $n$), $p\left(\mathbf{x}_2 \left| \mathbf{x}_1 \right.\right)$ (equal to $p\left(\mathbf{x}_{n+1} \left| \mathbf{x}_n \right.\right)$ for any $n$), and $p\left(\mathbf{y}_1 \left| \mathbf{x}_1 \right.\right)$ (equal to $p\left(\mathbf{y}_n \left| \mathbf{x}_n \right.\right)$ for any $n$).

The novelty of this work is to approximate the distribution $p\left(\mathbf{x}_1, \mathbf{y}_1, \mathbf{x}_2, \mathbf{y}_2\right)$ by a mixture with $K^2$ components, in such a way that the switching model defined by the mixture would allow a fast exact optimal filter. Such an approximation had limited interest until recently as it is well known that exact filters do not exist in classical switching models [1–3]. However, a class of switching models, called "conditionally Markov switching hidden linear models" (CMSHLMs), in which exact fast filters do exist, has been proposed recently in [4], and thus the idea is to use them.

More precisely, consider the "mixture approximation"

$$p\left(\mathbf{x}_1^2, \mathbf{y}_1^2\right) = \sum_{1 \le i,j, \le K} \alpha_{i,j} \; p_{i,j}(\mathbf{x}_1^2, \mathbf{y}_1^2). \quad (3)$$

Coefficient $\alpha_{i,j}$ can be interpreted as a distribution $\alpha_{i,j} = P(R_1 = i, R_2 = j)$ of a couple of random variables $(R_1, R_2)$ taking their values in $\Omega = \{1, \ldots, K\}$. The novelty is to consider the stationary triplet Markov chain $\mathbf{T}_1^N = (\mathbf{X}_1^N, \mathbf{R}_1^N, \mathbf{Y}_1^N)$, with $\mathbf{R}_1^N = (R_1, \ldots, R_N)$, whose distribution is defined by

$$p\left(\mathbf{x}_1^2, \mathbf{r}_1^2, \mathbf{y}_1^2\right) = \alpha_{r_1,r_2} \; p_{r_1,r_2}(\mathbf{x}_1^2, \mathbf{y}_1^2), \quad (4)$$

which would belong to the CMSHLM family [4].

In this paper, we propose to use suitable Gaussian distributions $p_{i,j}(\mathbf{x}_1^2, \mathbf{y}_1^2)$ and show that the exact filtering in approximate model works well under very weak hypotheses. In fact, all we need is to be able to sample realizations of $(\mathbf{X}_1^N, \mathbf{Y}_1^N)$ according to (1), (2). More precisely, once such a sample is simulated, we consider it as being a sample of a hidden discrete Markov chain $(\mathbf{R}_1^N, \mathbf{Z}_1^N)$, with $\mathbf{R}_1^N$ hidden and $\mathbf{Z}_1^N = (\mathbf{X}_1^N, \mathbf{Y}_1^N)$ observed, and we estimate the mixture parameters of interest via a kind of "Expectation-Maximization" (EM) algorithm. Our method is tested, and provides interesting results with respect to the classic Particle Filter (PF), on the following classic stochastic volatility model [3, 5]:

$$\mathbf{X}_{n+1} = \mu + \phi(\mathbf{X}_n - \mu) + \sigma \mathbf{U}_n; \quad (5)$$
$$\mathbf{Y}_{n+1} = \beta \exp(\mathbf{X}_{n+1}/2) \mathbf{V}_{n+1}, \quad (6)$$

with $\mathbf{U}_1$, $\mathbf{V}_1$, $\ldots$, $\mathbf{U}_N$, $\mathbf{V}_N$ independent centred Gaussian variables with variance 1.

Our method appears as an alternative to the widely used particle filter based methods, in particular applied in economics and finance [3, 5–7] or tracking [2, 3]. The paper is organized as follows. In next Section, we first recall the particular switching model studied in [8], which belongs to the CMSHLMs family [4], and thus which allows one to perform fast exact optimal filtering. The mixture estimation, which provides the approximation (3), is specified in Section 3. Fourth Section contains experiments and the last one draws conclusions and perspectives.

## 2. EXACT FILTERING IN CONDITIONALLY MARKOV SWITCHING HIDDEN LINEAR MODELS

Let us consider three random sequences $\mathbf{X}_1^N$, $\mathbf{R}_1^N$, and $\mathbf{Y}_1^N$, as specified in Introduction. Both $\mathbf{X}_1^N$ and $\mathbf{R}_1^N$ are hidden, while $\mathbf{Y}_1^N$ is observed. The process $\mathbf{R}_1^N$ can be seen as modelling the random "switches" of the distributions linked with $(\mathbf{X}_1^N, \mathbf{Y}_1^N)$. The "optimal filter" problem we deal with in this section consists of the sequential search of $(\mathbf{R}_1^N, \mathbf{X}_1^N)$ from $\mathbf{Y}_1^N$. More precisely, we search $p\left(r_{n+1}\left|\mathbf{y}_1^{n+1}\right.\right)$, $\mathrm{E}\left[\mathbf{X}_{n+1}\left|r_{n+1}, \mathbf{y}_1^{n+1}\right.\right]$ and $\mathrm{E}\left[\mathbf{X}_{n+1}\mathbf{X}_{n+1}^T\left|r_{n+1}, \mathbf{y}_1^{n+1}\right.\right]$ from $p\left(r_n\left|\mathbf{y}_1^n\right.\right)$, $\mathrm{E}\left[\mathbf{X}_n\left|r_n, \mathbf{y}_1^n\right.\right]$, $\mathrm{E}\left[\mathbf{X}_n\mathbf{X}_n^T\left|r_n, \mathbf{y}_1^n\right.\right]$ and $\mathbf{y}_{n+1}$. The optimal filter is then given by $\mathrm{E}\left[\mathbf{X}_{n+1}\left|\mathbf{y}_1^{n+1}\right.\right] = \sum_{r_{n+1}} p\left(r_{n+1}\left|\mathbf{y}_1^{n+1}\right.\right) \mathrm{E}\left[\mathbf{X}_{n+1}\left|r_{n+1}, \mathbf{y}_1^{n+1}\right.\right]$ and its variance $\mathrm{Var}\left[\mathbf{X}_{n+1}\left|\mathbf{y}_1^{n+1}\right.\right]$obtained from

$$\mathrm{E}\left[\mathbf{X}_{n+1}\mathbf{X}_{n+1}^T\left|\mathbf{y}_1^{n+1}\right.\right] = \sum_{r_{n+1}} p\left(r_{n+1}\left|\mathbf{y}_1^{n+1}\right.\right) \mathrm{E}\left[\mathbf{X}_{n+1}\mathbf{X}_{n+1}^T\left|r_{n+1}, \mathbf{y}_1^{n+1}\right.\right].$$

Such a problem is of importance in numerous situations and hundreds papers deal with different solutions for several decades. Usually, the distribution of $\mathbf{T}_1^N = (\mathbf{X}_1^N, \mathbf{R}_1^N, \mathbf{Y}_1^N)$ is defined assuming the Markovianity of $(\mathbf{X}_1^N, \mathbf{R}_1^N)$, and then defining the distribution of $\mathbf{Y}_1^N$ conditional on $(\mathbf{X}_1^N, \mathbf{R}_1^N)$ as being of the form $p\left(\mathbf{y}_1^N\left|r_1^N, \mathbf{x}_1^N\right.\right) = \prod_{n=1}^N p\left(\mathbf{y}_n\left|r_n, \mathbf{x}_n\right.\right)$. Fast exact filtering has not been proposed in such models until now, the problem coming from the fact that the distributions $p\left(r_n\left|\mathbf{y}_1^n\right.\right)$ are not easy to compute. A different model, called "Conditionally Markov Switching Hidden Linear Model" (CMSHLM), in which fast exact filtering is possible, has been recently proposed in [4] (see also its comparison with classic models in [9]). Here we will consider its particular Gaussian form studied in [8], which will be called "Stationary Conditionally Gaussian Observed Markov Switching Model" (SCGOMSM).

First, we will assume that

$$p\left(r_{n+1}\left|\mathbf{x}_1^n, \mathbf{r}_1^n, \mathbf{y}_1^n\right.\right) = p\left(r_{n+1}\left|r_n\right.\right). \tag{7}$$

In particular, this implies that $\mathbf{R}_1^N$ is a Markov chain. Let us set $\mathbf{Z}_n = (\mathbf{X}_n, \mathbf{Y}_n)^T$, and $\mathbf{W}_n = (\mathbf{U}_n, \mathbf{V}_n)^T$ (Gaussian white noise), $\Gamma_{\mathbf{Z}_n} = Cov(\mathbf{Z}_n, \mathbf{Z}_n^T)^T$. Let us consider a stationary $\mathbf{T}_1^N$, with Gaussian $p_{\mathbf{r}_1^2}(\mathbf{x}_1^2, \mathbf{y}_1^2)$ in (4). Thus we can say that the distribution of $\mathbf{T}_1^N$ is defined by $p\left(\mathbf{r}_1^2\right)$ (with equal margins) and Gaussian distributions given by means and covariance matrices

$$\mathbf{M}^{\mathbf{Z}}(r_1) = \begin{bmatrix}\mathbf{M}^{\mathbf{X}}(r_1)\\ \mathbf{M}^{\mathbf{Y}}(r_1)\end{bmatrix} = \mathrm{E}\left[\mathbf{Z}_1\left|R_1 = r_1\right.\right], \tag{8}$$

$$\Gamma^{\mathbf{Z}_1,\mathbf{Z}_2}(r_1, r_2) = \mathrm{E}\left[(\mathbf{Z}_1 - \mathbf{M}^{\mathbf{Z}}(r_1))(\mathbf{Z}_2 - \mathbf{M}^{\mathbf{Z}}(r_2))^T\right]. \tag{9}$$

Then, setting

$$\Gamma^{\mathbf{Z}_1,\mathbf{Z}_2}(\mathbf{r}_1^2) = \begin{bmatrix}\Gamma_{\mathbf{Z}_1}(r_1) & \Gamma^T_{\mathbf{Z}_1,\mathbf{Z}_2}(\mathbf{r}_1^2)\\ \Gamma_{\mathbf{Z}_1,\mathbf{Z}_2}(\mathbf{r}_1^2) & \Gamma_{\mathbf{Z}_2}(r_2)\end{bmatrix}, \tag{10}$$

$$\mathbf{A}(\mathbf{r}_1^2) = \Gamma_{\mathbf{Z}_1,\mathbf{Z}_2}(\mathbf{r}_1^2)\, \Gamma^{-1}_{\mathbf{Z}_1}(r_1), \tag{11}$$

and considering $\mathbf{B}(\mathbf{r}_1^2)\mathbf{B}^T(\mathbf{r}_1^2)$ such that

$$\mathbf{B}(\mathbf{r}_1^2)\mathbf{B}^T(\mathbf{r}_1^2) = \Gamma_{\mathbf{Z}_2}(r_2) - \Gamma_{\mathbf{Z}_1,\mathbf{Z}_2}(\mathbf{r}_1^2)\, \Gamma^{-1}_{\mathbf{Z}_1}(r_1)\, \Gamma^T_{\mathbf{Z}_1,\mathbf{Z}_2}(\mathbf{r}_1^2), \tag{12}$$

we can state that the distribution of $\mathbf{T}_1^N = (\mathbf{X}_1^N, \mathbf{R}_1^N, \mathbf{Y}_1^N)$ is defined by the following conditions

$$\mathbf{R}_1^N \text{ is a Markov chain;} \tag{13}$$

$$\mathbf{Z}_{n+1} = \mathbf{A}(\mathbf{R}_n^{n+1})(\mathbf{Z}_n - \mathbf{M}^{\mathbf{Z}}(R_n)) + \mathbf{B}(\mathbf{R}_n^{n+1})\mathbf{W}_{n+1} + \mathbf{M}^{\mathbf{Z}}(R_{n+1}) \tag{14}$$

with $\mathbf{A}$ and $\mathbf{B}$ verifying (11), (12), and $(\mathbf{W}_n)$ a Gaussian white noise with identity covariance matrix.

For reasons specified in Remark 1 below, we will consider that $\mathbf{A}(\mathbf{r}_n^{n+1})$ has the following form

$$\mathbf{A}(\mathbf{r}_n^{n+1}) = \begin{bmatrix}\mathbf{A}^1(\mathbf{r}_n^{n+1}) & 0\\ 0 & \mathbf{A}^4(\mathbf{r}_n^{n+1})\end{bmatrix}. \tag{15}$$

Then the model (7)-(15) is a particular CMSHLM and thus allows exact fast filtering, whose exact run is specified below.

**Remark 1** As shown in [8], model (7)-(15) is close to the classic "Conditionally Gaussian Linear State Space Model" (CGLSSM) [1, 3], whose general form is:

$$\begin{aligned}
&\mathbf{R}_1^N \text{ is a Markov chain;}\\
&\mathbf{X}_{n+1} = \mathbf{C}^1_{n+1}(R_{n+1})(\mathbf{X}_n - \mathbf{M}^{\mathbf{X}}(R_n)) + \mathbf{C}^2_{n+1}(R_{n+1})\mathbf{U}_{n+1} + \mathbf{M}^{\mathbf{X}}(R_{n+1})\\
&\mathbf{Y}_{n+1} = \mathbf{C}^3_{n+1}(R_{n+1})(\mathbf{X}_n - \mathbf{M}^{\mathbf{X}}(R_n)) + \mathbf{C}^4_{n+1}(R_{n+1})\mathbf{V}_{n+1} + \mathbf{M}^{\mathbf{Y}}(R_{n+1}).
\end{aligned}$$

Let us notice that in the model (7)-(15), $p\left(\mathbf{x}_1^N\left|r_1^N, \mathbf{y}_1^N\right.\right)$ is of (1)-(2) kind, which has been chosen on purpose to be weel-suited to this classic form. However, other families of

Gaussian distributions could have been considered. More precisely, any $\mathbf{A}(\mathbf{r}_n^{n+1})$ of the form (16) would still allow exact filtering [9]

$$\mathbf{A}(\mathbf{r}_n^{n+1}) = \begin{bmatrix} \mathbf{A}^1(\mathbf{r}_n^{n+1}) & \mathbf{A}^2(\mathbf{r}_n^{n+1}) \\ 0 & \mathbf{A}^4(\mathbf{r}_n^{n+1}) \end{bmatrix}. \tag{16}$$

For latter use, from (14) and (15), let

$$\mathbf{N}^{\mathbf{X}}(\mathbf{r}_n^{n+1}) = \mathbf{M}^{\mathbf{X}}(r_{n+1}) - \mathbf{A}^1(\mathbf{r}_n^{n+1})\mathbf{M}^{\mathbf{X}}(r_n),$$
$$\mathbf{N}^{\mathbf{Y}}(\mathbf{r}_n^{n+1}) = \mathbf{M}^{\mathbf{Y}}(r_{n+1}) - \mathbf{A}^4(\mathbf{r}_n^{n+1})\mathbf{M}^{\mathbf{Y}}(r_n),$$

and $\mathbf{Q}(\mathbf{R}_n^{n+1}) = \mathbf{B}(\mathbf{r}_n^{n+1})\mathbf{B}^T(\mathbf{r}_n^{n+1})$

$$\mathbf{Q}(\mathbf{r}_n^{n+1}) = \begin{bmatrix} \mathbf{Q}^1(\mathbf{r}_n^{n+1}) & \mathbf{Q}^2(\mathbf{r}_n^{n+1}) \\ \mathbf{Q}^3(\mathbf{r}_n^{n+1}) & \mathbf{Q}^4(\mathbf{r}_n^{n+1}) \end{bmatrix}.$$

Let us specify how the exact filtering runs. As the proposed method remains valid in any case of mixtures (3) once (4) defined a CMSHLM, we will first briefly recall the definition of a CMSHLM and then specify how the filter runs. A CMSHLM verifies:

$\mathbf{T}_1^N = (\mathbf{X}_1^N, \mathbf{R}_1^N, \mathbf{Y}_1^N)$ is Markov with

$$p\left(r_{n+1}, \mathbf{y}_{n+1} \left| \mathbf{x}_n, r_n, \mathbf{y}_n \right.\right) = p\left(r_{n+1}, \mathbf{y}_{n+1} \left| r_n, \mathbf{y}_n \right.\right); \tag{17}$$

$$\mathbf{X}_{n+1} = \mathbf{F}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})\mathbf{X}_n + \mathbf{G}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})\mathbf{W}_{n+1} + \mathbf{H}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1}) \tag{18}$$

with $\mathbf{F}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})$, $\mathbf{G}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})$ appropriate matrices, $\mathbf{W}_{n+1}$ appropriate white noise, and $\mathbf{H}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})$ appropriate vectors. $p\left(r_{n+1} \left| \mathbf{y}_1^{n+1} \right.\right)$, $\mathrm{E}\left[\mathbf{X}_{n+1} \left| r_{n+1}, \mathbf{y}_1^{n+1} \right.\right]$ and $\mathrm{E}\left[\mathbf{X}_{n+1}\mathbf{X}_{n+1}^T \left| r_{n+1}, \mathbf{y}_1^{n+1} \right.\right]$ can then be computed from $p\left(r_{n+1}, \mathbf{y}_{n+1} \left| r_n, \mathbf{y}_n \right.\right)$, $\mathbf{F}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})$, $\mathbf{H}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})$, $p\left(r_n \left| \mathbf{y}_1^n \right.\right)$ and $\mathrm{E}\left[\mathbf{X}_n \left| r_n, \mathbf{y}_1^n \right.\right]$ with complexity independent from $n$ as follows:

$$p\left(r_{n+1} \left| \mathbf{y}_1^{n+1} \right.\right) = \frac{\sum_{r_n} p\left(r_{n+1}, \mathbf{y}_{n+1} \left| r_n, \mathbf{y}_n \right.\right) p\left(r_n \left| \mathbf{y}_1^n \right.\right)}{\sum_{r_n, r_{n+1}^*} p\left(r_{n+1}^*, \mathbf{y}_{n+1} \left| r_n, \mathbf{y}_n \right.\right) p\left(r_n \left| \mathbf{y}_1^n \right.\right)}, \tag{19}$$

$$p\left(r_n \left| r_{n+1}, \mathbf{y}_1^{n+1} \right.\right) = \frac{p\left(r_{n+1}, \mathbf{y}_{n+1} \left| r_n, \mathbf{y}_n \right.\right) p\left(r_n \left| \mathbf{y}_1^n \right.\right)}{\sum_{r_n^*} p\left(r_{n+1}, \mathbf{y}_{n+1} \left| r_n^*, \mathbf{y}_n \right.\right) p\left(r_n^* \left| \mathbf{y}_1^n \right.\right)} \tag{20}$$

$$\mathrm{E}\left[\mathbf{X}_{n+1} \left| r_{n+1}, \mathbf{y}_1^{n+1} \right.\right] = \sum_{r_n} p\left(r_n \left| r_{n+1}, \mathbf{y}_1^{n+1} \right.\right) \Big(\mathbf{F}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})\mathrm{E}\left[\mathbf{X}_n \left| r_n, \mathbf{y}_1^n \right.\right] + \mathbf{H}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})\Big), \tag{21}$$

and

$$\mathrm{E}\left[\mathbf{X}_{n+1}\mathbf{X}_{n+1}^T \left| r_{n+1}, \mathbf{y}_1^{n+1} \right.\right] = \sum_{r_n} p\left(r_n \left| r_{n+1}, \mathbf{y}_1^{n+1} \right.\right) \Big(\mathbf{F}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})\mathrm{E}\left[\mathbf{X}_n\mathbf{X}_n^T \left| r_n, \mathbf{y}_1^n \right.\right]\mathbf{F}^T(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1}) + \mathbf{F}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})\mathrm{E}\left[\mathbf{X}_n \left| r_n, \mathbf{y}_1^n \right.\right]\mathbf{H}^T(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1}) + \mathbf{H}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})\mathrm{E}^T\left[\mathbf{X}_n \left| r_n, \mathbf{y}_1^n \right.\right]\mathbf{F}^T(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1}) + \mathbf{G}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})\mathbf{G}^T(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})\Big). \tag{22}$$

Let us now verify that SCGOMSM $\mathbf{T}_1^N$ defined by $p(r_1, r_2)$ and (7)-(15) is a CMSHLM (17)-(18). Let us first verify (17). According to (7) we have $p\left(r_{n+1} \left| \mathbf{x}_n, r_n, \mathbf{y}_n \right.\right) = p\left(r_{n+1} \left| r_n \right.\right)$ and, according to (14)-(15), we have $p\left(\mathbf{y}_{n+1} \left| \mathbf{x}_n, \mathbf{y}_n, \mathbf{r}_n^{n+1} \right.\right) = p\left(\mathbf{y}_{n+1} \left| \mathbf{y}_n, \mathbf{r}_n^{n+1} \right.\right)$. These two equalities give

$$p\left(r_{n+1}, \mathbf{y}_{n+1} \left| \mathbf{x}_n, r_n, \mathbf{y}_n \right.\right) = p\left(r_{n+1}, \mathbf{y}_{n+1} \left| r_n, \mathbf{y}_n \right.\right).$$

Let us now verify (18). According to (14) the distribution $p\left(\mathbf{x}_{n+1}, \mathbf{y}_{n+1} \left| \mathbf{x}_n, \mathbf{r}_n^{n+1}, \mathbf{y}_n \right.\right)$ is Gaussian with mean

$$\begin{bmatrix} \mathbf{A}^1(\mathbf{r}_n^{n+1})\mathbf{x}_n + \mathbf{N}^{\mathbf{X}}(\mathbf{r}_n^{n+1}) \\ \mathbf{A}^4(\mathbf{r}_n^{n+1})\mathbf{y}_n + \mathbf{N}^{\mathbf{Y}}(\mathbf{r}_n^{n+1}) \end{bmatrix}$$

and variance $\mathbf{Q}(\mathbf{r}_n^{n+1})$. Using the classical Gaussian conditioning rules we can say that the distribution $p\left(\mathbf{x}_{n+1} \left| \mathbf{x}_n, \mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1} \right.\right)$ is then Gaussian with mean

$$\mathbf{A}^1(\mathbf{r}_n^{n+1})\mathbf{x}_n + \mathbf{N}^{\mathbf{X}}(\mathbf{r}_n^{n+1}) + \mathbf{Q}^2(\mathbf{r}_n^{n+1})(\mathbf{Q}^4(\mathbf{r}_n^{n+1}))^{-1} \Big(\mathbf{y}_{n+1} - \mathbf{A}^4(\mathbf{r}_n^{n+1})\mathbf{y}_n - \mathbf{N}^{\mathbf{Y}}(\mathbf{r}_n^{n+1})\Big),$$

and variance

$$\mathbf{Q}^1(\mathbf{r}_n^{n+1}) - \mathbf{Q}^2(\mathbf{r}_n^{n+1})(\mathbf{Q}^4(\mathbf{r}_n^{n+1}))^{-1}\mathbf{Q}^3(\mathbf{r}_n^{n+1}).$$

Then we can state, according to classic properties of Gaussian laws, that (18) is verified with

$$\begin{aligned} \mathbf{F}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1}) &= \mathbf{A}^1(\mathbf{r}_n^{n+1}), \\ \mathbf{H}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1}) &= \mathbf{Q}^2(\mathbf{r}_n^{n+1})(\mathbf{Q}^4(\mathbf{r}_n^{n+1}))^{-1} \\ (\mathbf{Y}_{n+1} - \mathbf{A}^4(\mathbf{r}_n^{n+1})\mathbf{Y}_n &- \mathbf{N}^{\mathbf{Y}}(\mathbf{r}_n^{n+1})) + \mathbf{N}^{\mathbf{X}}(\mathbf{r}_n^{n+1}), \\ \mathbf{G}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})\mathbf{G}^T(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1}) &= \\ \mathbf{Q}^1(\mathbf{r}_n^{n+1}) - \mathbf{Q}^2(\mathbf{r}_n^{n+1})&(\mathbf{Q}^4(\mathbf{r}_n^{n+1}))^{-1}\mathbf{Q}^3(\mathbf{r}_n^{n+1}). \end{aligned} \tag{23}$$

Finally, for given $\mathbf{M}^{\mathbf{Z}}(r_n)$, $\Gamma^{\mathbf{Z}_1,\mathbf{Z}_2}(\mathbf{r}_n^{n+1})$, $p\left(r_n \left| \mathbf{y}_1^n \right.\right)$, $\mathrm{E}\left[\mathbf{X}_n \left| r_n, \mathbf{y}_1^n \right.\right]$, and $\mathbf{y}_{n+1}$, the optimal filter in the system (7)-(15) is

**Algorithm 1**

(i) consider $\mathbf{M}^{\mathbf{Z}}(r_n)$ and $\Gamma^{\mathbf{Z}_1,\mathbf{Z}_2}(\mathbf{r}_n^{n+1})$ verifying (8), (9);
(ii) compute $\mathbf{Q}(\mathbf{r}_n^{n+1})$ using (12);
(iii) compute $\mathbf{F}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})$, $\mathbf{H}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})$ and $\mathbf{G}(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})\mathbf{G}^T(\mathbf{r}_n^{n+1}, \mathbf{y}_n^{n+1})$ with (23);
(iv) compute $p\left(r_{n+1}, \mathbf{y}_{n+1} \left| r_n, \mathbf{y}_n \right.\right) = p\left(r_{n+1} \left| r_n \right.\right) p\left(\mathbf{y}_{n+1} \left| \mathbf{r}_n^{n+1}, \mathbf{y}_n \right.\right)$, knowing that $p\left(\mathbf{y}_{n+1} \left| \mathbf{r}_n^{n+1}, \mathbf{y}_n \right.\right)$ is Gaussian with mean $\mathbf{A}^4(\mathbf{r}_n^{n+1})\mathbf{y}_n + \mathbf{N}^{\mathbf{Y}}(\mathbf{r}_n^{n+1})$ and variance $\mathbf{Q}^4(\mathbf{r}_n^{n+1})$;
(v) compute $p\left(r_{n+1} \left| \mathbf{y}_1^{n+1} \right.\right)$, $\mathrm{E}\left[\mathbf{X}_{n+1} \left| r_{n+1}, \mathbf{y}_1^{n+1} \right.\right]$ and $\mathrm{E}\left[\mathbf{X}_{n+1}\mathbf{X}_{n+1}^T \left| r_{n+1}, \mathbf{y}_1^{n+1} \right.\right]$ with (19)-(22).

## 3. APPROXIMATING STOCHASTIC VOLATILITY MODELS WITH CMSHLM

The idea is to simulate data $\mathbf{Z}_1^N = (\mathbf{X}_1^N, \mathbf{Y}_1^N) = (\mathbf{x}_1^N, \mathbf{x}_1^N)$ with a non-linear and non-Gaussian model (1)-(2) and then to consider them as generated by a model $\mathbf{T}_1^N = (\mathbf{X}_1^N, \mathbf{R}_1^N, \mathbf{Y}_1^N)$ verifying (7)-(15). Therefore the sampled data $\mathbf{z}_1^N = (\mathbf{x}_1^N, \mathbf{y}_1^N)$ are considered as produced by a classic Gaussian hidden Markov model $(\mathbf{R}_1^N, \mathbf{Z}_1^N)$. The model identification problem is then an HMC estimation problem and there exist several methods to deal with. We chose to use a method involving the classic Expectation-Maximization (EM) algorithm, knowing that EM proved to be quite efficient in Gaussian cases we deal with. More precisely, we propose the following:

**Algorithm 2**

(i) from simulated $\mathbf{z}_1^{N_{EM}}$, estimate parameters of the Gaussian hidden Markov model $(\mathbf{R}_1^{N_{EM}}, \mathbf{Z}_1^{N_{EM}})$ with EM (see paragraph *Parameters estimation with EM* below);

(ii) use the estimated parameters to estimate $\mathbf{R}_1^{N_{EM}} = \hat{r}_1^{N_{EM}}$ with the Maximum Posterior Mode (MPM);

(iii) use $\hat{r}_1^{N_{EM}}$ and $\mathbf{z}_1^{N_{EM}}$ to estimate the complementary covariances $\Gamma_{\mathbf{Z}_1,\mathbf{Z}_2}(\mathbf{r}_1^2)$ in (10).

Finally, the whole filter proposed runs as follows:

**Main algorithm**

(a) use **Algorithm 2** to estimate the parameters of the corresponding SCGOMSM;

(b) simulate data $\mathbf{z}_1^N$ according to model (1), (2) considered, and use **Algorithm 1** to perform the fast filter.

Of course, sampled data $\mathbf{z}_1^{N_EM}$ used in **Algorithm 2** are not the data $\mathbf{z}_1^N$ used in the filtering.

**Parameters estimation with EM**

Let us briefly recall how the classic EM algorithm runs in simplified model (i), Algorithm 2. Let $\mathbf{Z}_1^N = (\mathbf{X}_1^N, \mathbf{Y}_1^N)$ and $\mathbf{T}_1^N = (\mathbf{R}_1^N, \mathbf{Z}_1^N)$. For known parameters forward probabilities $\alpha(r_n) = p(r_n, \mathbf{z}_1^n)$ and backward ones $\beta(r_n) = p\left(\mathbf{z}_{n+1}^N \,|r_n\right)$ are computed recursively with $\alpha(r_1) = p(t_1)$, $\alpha(r_{n+1}) = \sum_{r_n \in \Omega} \alpha(r_n) p\left(t_{n+1}\,|t_n\right)$ for $1 \leq n \leq N-1$; $\beta(r_N) = 1$, $\beta(r_n) = \sum_{r_{n+1}\in\Omega} \beta(r_{n+1}) p\left(t_{n+1}\,|t_n\right)$ for $1 \leq n \leq N-1$.

Then we have

$$p\left(\mathbf{r}_n^{n+1}\,|\mathbf{z}_1^N\right) = \frac{\alpha(r_n) p\left(t_{n+1}\,|t_n\right) \beta(r_{n+1})}{\sum\limits_{r_n^* \in \Omega} \alpha(r_n^*)\beta(r_n^*)},$$

and thus

$$p\left(r_n\,|\mathbf{z}_1^N\right) = \frac{\alpha(r_n)\beta(r_n)}{\sum\limits_{r_n^* \in \Omega} \alpha(r_n^*)\beta(r_n^*)}.$$

The parameters, set in a vector $\theta$, are for $i, j \in 1, \ldots, K$: $p_{i,j} = p(R_1 = i, R_2 = j)$, $\mathbf{M}^{\mathbf{Z}}(i) = \mathrm{E}\left[\mathbf{Z}_1\,|R_1 = i\right]$, and $\Gamma_{\mathbf{Z}}(i) = \mathrm{Var}\left[\mathbf{Z}\,|R_1 = i\right]$.

**Table 1**. Four cases corresponding to four different values for $\Phi$ ($\sigma^2 = 1 - \Phi^2$, $\mu = 0.5$, $\beta = 0.5$). Mean square error obtained with the new method (for $K = 2$ to 7 classes), and with the particle filter method. The time of restoration for $N = 1000$ data is given between parentheses (time in hundredth of second).

| # | Φ | 2 | 3 | 5 | 7 | PF |
|---|---|---|---|---|---|---|
| **1** | **0.99** | 0.45 (2.76) | 0.30 (4.15) | 0.24 (10.88) | 0.22 (20.73) | 0.21 (52.78) |
| **2** | **0.90** | 0.57 (2.83) | 0.50 (4.29) | 0.47 (10.86) | 0.47 (20.98) | 0.46 (53.86) |
| **3** | **0.80** | 0.65 (2.77) | 0.59 (4.27) | 0.58 (10.80) | 0.57 (21.38) | 0.57 (53.33) |
| **4** | **0.50** | 0.75 (2.97) | 0.71 (4.41) | 0.70 (10.79) | 0.70 (19.63) | 0.70 (52.60) |

Let $\Psi_n^{(q)}(i,j) = p\left(r_n = i, r_{n+1} = j\,\middle|\mathbf{z}_1^N, \theta^{(q)}\right)$ and $\phi_n^{(q)}(i) = p\left(r_n = i\,\middle|\mathbf{z}_1^N, \theta^{(q)}\right)$. EM is an iterative method, which produces a sequence $\theta^{(0)}, \ldots, \theta^{(q)}, \ldots$ in the following way. Consider $\theta^{(0)}$ found in some ways, calculate $\theta^{(q+1)}$ from $\theta^{(q)}$ by

$$p_{i,j}^{(q+1)} = \frac{\sum\limits_{n=1}^{N} \Psi_n^{(q)}(i,j)}{\sum\limits_{n=1}^{N} \phi_n^{(q)}(i)}; \mathbf{M}^{\mathbf{Z}}(i)^{(q+1)} = \frac{\sum\limits_{n=1}^{N} \mathbf{z}_n \phi_n^{(q)}(i)}{\sum\limits_{n=1}^{N} \phi_n^{(q)}(i)};$$

$$\Gamma_{\mathbf{Z}}(i)^{(q+1)} = \sum_{n=1}^{N} \Big[ \left(\mathbf{z}_n - \mathbf{M}^{\mathbf{Z}}(i)^{(q+1)}\right) \left(\mathbf{z}_n - \mathbf{M}^{\mathbf{Z}}(i)^{(q+1)}\right)^T \phi_n^{(q)}(i) \Big] / \sum_{n=1}^{N} \phi_n^{(q)}(i).$$

Stop iterations according to some criterion.

## 4. EXPERIMENTS

Let us consider the stochastic volatility model (5), (6) as an example of state-space system (1), (2). We present in Table 1 results of four experiments with different values for $\Phi$ and $\sigma^2$, with $\mu = 0.5$, $\beta = 0.5$ (in all cases $\mathrm{Var}\left[\mathbf{X}_n\right] = 1$, so that $\sigma^2 = 1 - \Phi^2$). Then we applied the whole filter with varying number of classes, from $K = 2$ to $K = 7$. Comparison is also performed with the classic Particle Filter (PF)[1] (1500 particles and Sequantial Importance Resampling).

The number of EM iterations was set to 100, $N^{EM} = 20000$ and $N = 1000$. The Means Square Error (MSE) results correspond to the means of 100 independent experiments. When using 4 or 5 values for each $r_n$ the results are comparable to those obtained with PF. When choosing 1500

[1] Algorithm was taken from http://www.ece.sunysb.edu/~zyweng/particle.html

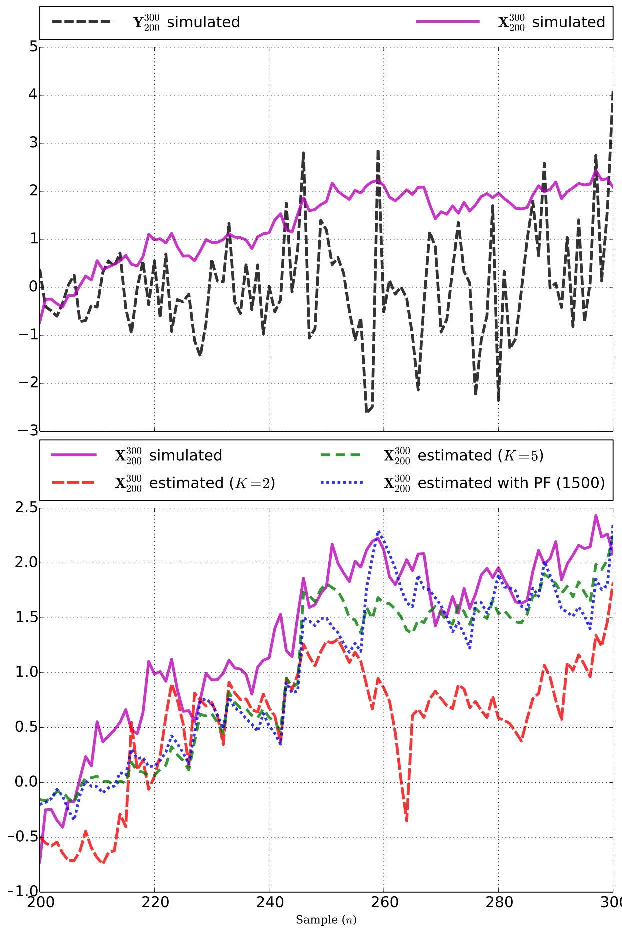


**Fig. 1**. Case 1 in Table 1. Up: trajectory of the hidden signal (magenta) and the noisy one (black). Down: real signal (magenta), proposed filter results for $K = 2$ classes (blue, dashed), and $K = 5$ classes (green, dotted).

particles, PF is faster than our method because of the EM algorithm; however, once the parameters learnt, our method is as fast as the classic Kalman filter, and faster than the PF (see Table 1). A restored trajectory corresponding to case 1 is presented in Figure 1.

Whatever SV parameters, the results remain comparable to the PF ones beyond 4 classes, which means that they are close to the optimal ones. The question of the automatic selection of an optimal value of $K$ for a given model is an open issue that should be further addressed.

## 5. CONCLUSION

We proposed a new method for the sequential search of a hidden signal in non-linear and non-Gaussian systems. The method is quite general and works under slight conditions; in fact, it is only required to be able to sample data according to the non-linear model considered. The method is based on the introduction of a Gaussian switching model, which approximate the system considered, and in which a fast exact optimal filtering is feasible. Once the parameters of the Gaussian switching model are estimated, the method is as complex as the classic Kalman filter. In addition, contrary to PF, it remains fast when the hidden realizations space and the observation one increase.

The method has been applied to a simple stochastic volatility model and it turns out that the MSE obtained is very close to the theoretical one, the latter having been estimated by the classical particle filter based method.

As perspectives, let us mention applications to other stochastic volatility models [4, 7, 8], applications to Bayesian tracking [3, 4], or the use of more complex families of switching models allowing fast exact filter. Parameter estimation allowing unsupervised filtering is another perspective.

## REFERENCES


[1] C. Andrieu, M. Davy, and A. Doucet, "Efficient particle filtering for jump Markov systems. Application to time-varying autoregressions," *IEEE Trans. on Signal Processing*, vol. 51, no. 7, pp. 1762–1770, Jul 2003.

[2] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb 2002.

[3] O. Cappé, E. Moulines, and T. Rydén, *Inference in Hidden Markov Models*, Springer-Verlag, 2005.

[4] W. Pieczynski, "Exact filtering in conditionally Markov switching hidden linear models," *Comptes Rendus de l'Académie des Sciences Mathématiques*, vol. 349, no. 9-10, pp. 587–590, 2011.

[5] D. Creal, "A survey of sequential Monte Carlo methods for economics and finance," *Econometric reviews*, vol. 31, no. 3, pp. 245–296, 2012.

[6] R. F. Engle and J. G. Rangel, "The spline-GARCH model for low-frequency volatility and its global macroeconomic causes," *Review of Financial Studies*, vol. 21, pp. 1187–1222, 2008.

[7] J. D. Hamilton and R. Susmel, "Autoregressive conditional heteroskedasticity and changes in regime," *Journal of Econometrics*, vol. 64, pp. 307–333, 1994.

[8] S. Derrode and W. Pieczynski, "Exact fast computation of optimal filter in Gaussian switching linear systems," *IEEE Signal Processing Letters*, vol. 20, no. 7, pp. 701–704, July 2013.

[9] N. Abbassi, D. Benboudjema, and W. Pieczynski, "Kalman filtering approximations in triplet Markov Gaussian switching models," in *Proc. IEEE Int. Workshop SSP'11*, Nice, France, June 2011, pp. 290–294.