
Filtrage statistique optimal rapide dans des systèmes linéaires à sauts non stationnaires

Noufel Abbassi¹, Stéphane Derrode², François Desbouvries¹, Yohan Petetin¹, Wojciech Pieczynski¹

1. Telecom Sudparis, Département CITI, CNRS UMR 5157, 9, rue Charles Fourier, 91000 Evry, France
2. Ecole Centrale de Lyon, CNRS, LIRIS, UMR 5205, 26 av. Guy de Collongue, 69130 Ecully, France

RESUME. Nous traitons du problème de filtrage statistique optimal dans des systèmes à sauts. Nous considérons trois processus : un processus continu caché X , un processus continu observé Y , et un processus discret caché R modélisant les « sauts », qui peuvent être vus comme les changements aléatoires des paramètres régissant localement les distributions markoviennes du couple (X, Y) . Nous nous intéressons à une famille récente de modèles dans laquelle il est possible de mettre en place un filtrage optimal rapide, dont la complexité est linéaire en temps. Nous étendons cette famille en introduisant un quatrième processus discret fini U permettant de modéliser les possibles non stationnarités du triplet (X, R, Y) . Nous montrons que les filtres optimaux rapides demeurent possibles dans la famille étendue et nous illustrons leur intérêt via quelques simulations.

ABSTRACT. This paper deals with optimal statistical filtering in jump systems. We consider three random sequences: a hidden real-valued process X , an observed real-valued process Y and a hidden discrete process R modeling jumps that can be interpreted as random switches in the parameters governing locally the Markovian distributions of the pairwise process (X, Y) . We focus on a recent family of models in which it is possible to implement a fast optimal filtering, whose complexity is linear in time. We extend this family by introducing a fourth hidden discrete process U to model possible non-stationarity in triplet (X, R, Y) . We show that fast optimal filtering remains possible in the extended family and illustrate their interest via some simulations.

MOTS-CLES : Système linéaire gaussien à sauts. Filtrage optimal exact. Modèle caché conditionnellement linéaire à sauts markoviens. Modèle caché conditionnellement linéaire à sauts marginalement markoviens.

KEYWORDS: Jump linear Gaussian system. Optimal and exact filtering. Conditionally Markov switching hidden linear model. Conditionally switching hidden linear model with marginally Markov jumps.

Extended abstract (non numéroté) : résumé anglais long

We consider triplet Markov Gaussian linear systems $T_1^N = (X_1^N, R_1^N, Y_1^N)$ where $X_1^N = (X_1, \dots, X_N)$ is a hidden continuous random sequence, $R_1^N = (R_1, \dots, R_N)$ is a hidden discrete Markov random sequence, $Y_1^N = (Y_1, \dots, Y_N)$ is an observed continuous random sequence, and (X_1^N, Y_1^N) is Markovian and Gaussian conditionally on R_1^N . Using usual notations for laws and conditional expectations, we seek to calculate $p(r_{n+1} | y_1^{n+1})$, $E[X_{n+1} | r_{n+1}, y_1^{n+1}]$ (which gives $E[X_{n+1} | y_1^{n+1}] = \sum_{r_{n+1}} p(r_{n+1} | y_1^{n+1}) E[X_{n+1} | r_{n+1}, y_1^{n+1}]$), and $Var[X_{n+1} | r_{n+1}, y_1^{n+1}]$, from $p(r_n | y_1^n)$, $E[X_n | r_n, y_1^n]$, $Var[X_n | r_n, y_1^n]$, and y_{n+1} . Process R_1^N can be interpreted as modeling the random switches, or jumps, of the parameters defining the law of (X_1^N, Y_1^N) .

In the classical “Conditionally Gaussian Linear State-Space Model” (CGLSSM) (Cappé *et al*, 2005), optimal filter is not workable with a reasonable complexity, and numerous approximation methods have been proposed. Among them particle filters are asymptotically optimal (Doucet *et al*, 2001) but present drawbacks such as the degeneration of weights and a relatively high computational burden.

In this paper, we propose a new model, quite close to the CGLSSM, belonging to the general recently proposed family of models, called “conditionally Markov switching hidden linear models” (CMSHLM) (Pieczynski,

2011a), in which the computation of optimal filter with complexity linear in the number of observations is possible. In a CMSHLM, the triplet $T_1^N = (X_1^N, R_1^N, Y_1^N)$ and the couple (R_1^N, Y_1^N) are Markovian, but the couple (X_1^N, R_1^N) is not necessarily. The Markovianity of (R_1^N, Y_1^N) plays a crucial role because it allows the rapid calculation of $p(r_{n+1}|y_1^{n+1})$ from $p(r_n|y_1^n)$ and y_n .

Then we show that it is possible to introduce a fourth discrete-valued process U_1^N , so that the triplet $T_1^N = (X_1^N, R_1^N, Y_1^N)$ is Markov. In such models, called ‘‘Conditionally switching hidden linear model with marginally Markov jumps’’, the couple (R_1^N, Y_1^N) is not necessarily Markovian. This context has already proven very effective in image segmentation for modeling multiple-stationaries (Lanchantin *et al.*, 2011; Boudaren *et al.*, 2012a), in hidden semi-Markov chains (Lapuyade-Lahorgue et Pieczynski, 2011a) or in hidden evidential Markov chains (Pieczynski, 2007 ; Boudaren *et al.*, 2012b ; Ramasso et Denoeux, 2013), but is novel in optimal statistical filtering. Simulations are provided to illustrate the value of the new modeling in the context of non-stationary on-line filtering of time-series.

1. Introduction

On considère trois séquences aléatoires $X_1^N = (X_1, \dots, X_N)$, $R_1^N = (R_1, \dots, R_N)$, et $Y_1^N = (Y_1, \dots, Y_N)$, où, pour $n = 1, \dots, N$, les variables X_n et Y_n prennent respectivement leurs valeurs dans \mathbb{R}^m et \mathbb{R}^q , alors que R_n prend ses valeurs dans un espace fini $S = \{1, \dots, K\}$. Les réalisations des séquences X_1^N et R_1^N sont cachées et seules les réalisations de Y_1^N sont observées. Le problème considéré est celui de l’estimation séquentielle optimale des réalisations de X_1^N et R_1^N à partir de Y_1^N . Plus précisément, on s’intéresse à l’estimation, pour tout $n = 1, \dots, N$, des réalisations de X_n et R_n à partir de celle de Y_1^n . Les méthodes étudiées sont optimales dans le sens où, d’une part, l’erreur quadratique moyenne entre la vraie réalisation de X_n et son estimée est minimale et, d’autre part, la probabilité de se tromper en estimant la réalisation de R_n est également minimale. La première estimée est obtenue par l’espérance conditionnelle, et la deuxième par la maximisation de la probabilité conditionnelle aux observations. L’estimation sera « séquentielle », ce qui signifie qu’afin d’obtenir des estimateurs fonctionnant avec des temps raisonnables, on recherchera les estimées $(\hat{x}_{n+1}, \hat{r}_{n+1})$ de $(X_{n+1}, R_{n+1}) = (x_{n+1}, r_{n+1})$ (obtenues à partir de y_1^{n+1}) en utilisant les estimées précédentes (\hat{x}_n, \hat{r}_n) de $(X_n, R_n) = (x_n, r_n)$ (obtenues à partir de y_1^n) et la nouvelle donnée y_{n+1} .

Finalement, avec les notations habituelles pour les lois et les espérances conditionnelles, on cherche à déterminer $p(r_{n+1}|y_1^{n+1})$, $E[X_{n+1}|r_{n+1}, y_1^{n+1}]$ (ce qui donne $E[X_{n+1}|y_1^{n+1}] = \sum_{r_{n+1}} p(r_{n+1}|y_1^{n+1})E[X_{n+1}|r_{n+1}, y_1^{n+1}]$), et $Var[X_{n+1}|r_{n+1}, y_1^{n+1}]$, à partir de $p(r_n|y_1^n)$, $E[X_n|r_n, y_1^n]$, $Var[X_n|r_n, y_1^n]$, et y_{n+1} .

De telles modélisations jouent un rôle important en situations fortement non stationnaires ; en effet, le processus R_1^N peut être interprété comme modélisant les « sauts » aléatoires des paramètres définissant la loi du couple (X_1^N, Y_1^N) . L’idée classique pour modéliser la loi du triplet $T_1^N = (X_1^N, R_1^N, Y_1^N)$ est de considérer simultanément deux modélisations, dont chacune a fait la preuve, dans différents domaines d’applications respectifs, de son efficacité. Le processus R_1^N est supposé markovien ; ensuite on définit la loi de X_1^N conditionnelle à R_1^N , en la considérant également comme markovienne. Le couple (X_1^N, R_1^N) est alors le modèle classique de « chaîne de Markov cachée ». Ensuite, conditionnellement à (X_1^N, R_1^N) , la loi de Y_1^N est très simple : les variables Y_1, \dots, Y_N sont considérées indépendantes et la loi de chaque Y_n conditionnelle à (X_1^N, R_1^N) est supposée ne dépendre que de (X_n, R_n) . Dans le cas le plus simple le processus R_1^N est markovien et la loi du couple (X_1^N, Y_1^N) conditionnelle à R_1^N est celle d’un système linéaire gaussien markovien. Un tel modèle est dit, en anglais, ‘‘Conditionally Gaussian Linear State-Space Model’’ et sera dans la suite désigné par CGLSSM (Cappé *et al.*, 2005). Il est défini par

$$R_1^N \text{ est de Markov avec } p(r_{n+1}|x_1^n, r_1^n, y_1^n) = p(r_{n+1}|r_n) ; \quad (1.1)$$

$$X_{n+1} = A_{n+1}(R_{n+1})X_n + C_{n+1}(R_{n+1})U_{n+1}; \quad (1.2)$$

$$Y_{n+1} = B_{n+1}(R_{n+1})X_{n+1} + D_{n+1}(R_{n+1})V_{n+1}, \quad (1.3)$$

avec $A_{n+1}(R_{n+1})$, $B_{n+1}(R_{n+1})$, $C_{n+1}(R_{n+1})$, $D_{n+1}(R_{n+1})$ des matrices de taille adéquate, dépendantes des sauts, le processus (U_1, V_1) , ..., (U_N, V_N) étant un bruit blanc gaussien (alors pour $n=1, \dots, N-1$ le vecteur (U_{n+1}, V_{n+1}) est indépendant de $T_1^n = (X_1^n, R_1^n, Y_1^n)$). Conditionnellement à R_1^N , un CGLSSM est ainsi un système linéaire gaussien markovien classique et, lorsque les sauts sont connus, le filtre de Kalman peut être utilisé pour résoudre le problème considéré. Cependant, lorsque les sauts ne sont pas observés, une solution du problème avec une complexité linéaire – ou même polynomiale – en nombre d'observations n n'a toujours pas été proposée. Diverses méthodes d'approximations, qui peuvent être déterministes (Costa *et al.*, 2005 ; Kim et Nelson, 1999 ; Zoeter et Heskes, 2006 ; Yin *et al.*, 2010) ou stochastiques (Andrieu *et al.*, 2003 ; Ristic *et al.*, 2004 ; Doucet *et al.*, 2001) ont alors été proposées et appliquées souvent avec succès. En particulier, les méthodes « Monte Carlo par Chaînes de Markov Séquentielles » (MCMCS), également connues sous le nom de « filtre particulaire » (FP), présentent des propriétés d'optimalité asymptotique et peuvent donc être considérées comme quasi-optimales dans un grand nombre de situations. En conséquence le FP est très couramment utilisé. Cependant, ces méthodes peuvent présenter un certain nombre d'inconvénients, comme la dégénérescence des poids pouvant impliquer l'accroissement du temps d'exécution. Les applications du filtrage optimal dans des systèmes à sauts sont multiples et touchent des domaines très variés. Citons, à titre d'exemple et de manière non exhaustive, la poursuite de cible (Arulampalam *et al.*, 2003), le traitement de la parole (Lee et Rheem, 2000 ; Soo *et al.*, 2012), la modélisation et le traitement de la volatilité (Carvalho *et al.*, 2007 ; Elliot et Miao, 2006 ; Yu et Zhang, 2011), le traitement d'images (Wu *et al.*, 2004 ; Smith *et al.*, 2010), modélisations de la production industrielle (Giordani *et al.*, 2007), propagation des épidémies (Zhou et Shumway, 2008), ou encore la modélisation et le traitement des différents phénomènes liés aux marchés (Al-Anaswah et Wilfling, 2011 ; Johnson et Sakoulis, 2008).

Plus récemment il a été proposé différents modèles « à sauts », dont certains très simples, dans lesquels le problème peut être résolu de manière exacte avec une complexité linéaire en n . Leur synthèse est présentée dans (Pieczynski et Abbassi, 2009 ; Abbassi, 2012). Ensuite, un modèle général, appelé « modèle caché conditionnellement linéaire à sauts markoviens » (MCCLSM), unifiant et étendant les différents modèles précédents, a été proposé dans (Pieczynski, 2011a). La différence entre la famille classique des modèles et le MCCLSM, dans lequel il est donc possible de mettre en place un filtrage optimal rapide présentant une complexité comparable à celle du filtre de Kalman dans des systèmes gaussiens classiques, est la suivante. Les hypothèses retenues pour définir la loi de $T_1^N = (X_1^N, R_1^N, Y_1^N)$ sont très proches, à condition de permuter X_1^N et Y_1^N . Dans les démarches classiques on suppose R_1^N markovien, ensuite on suppose markovien X_1^N conditionnellement à R_1^N , et on termine la construction de la loi de $T_1^N = (X_1^N, R_1^N, Y_1^N)$ en considérant une distribution simple de Y_1^N conditionnellement à (X_1^N, R_1^N) . On arrive aux modèles où le triplet $T_1^N = (X_1^N, R_1^N, Y_1^N)$ et le couple (X_1^N, R_1^N) sont markoviens, mais le couple (R_1^N, Y_1^N) ne l'est pas nécessairement. Dans un MCCLSM, le triplet $T_1^N = (X_1^N, R_1^N, Y_1^N)$ et le couple (R_1^N, Y_1^N) sont markoviens, sans que le couple (X_1^N, R_1^N) le soit nécessairement. La markovianité du couple (R_1^N, Y_1^N) joue un rôle crucial car elle permet le calcul rapide de $p(r_{n+1}|y_1^{n+1})$ à partir de $p(r_n|y_1^n)$ et y_n . Le point important, à l'origine du présent travail, est alors le suivant. Lorsque l'on a opté pour l'utilisation d'un MCCLSM on calcule d'abord $p(r_{n+1}|y_1^{n+1})$, et ensuite $E[X_{n+1}|y_1^{n+1}]$. Le premier calcul étant indépendant du second, tout se passe comme si on était confronté au problème de la recherche de R_1^N à partir de Y_1^N , qui est le problème de segmentation et qui a été étudié dans les cadres des différentes extensions de la loi classique – typiquement celle d'un Markov caché – du couple (R_1^N, Y_1^N) . Il est ainsi possible d'utiliser ces différentes extensions, qui se sont souvent montrées d'un remarquable intérêt, dans le cadre des systèmes à sauts considérés dans ce travail. Plus précisément, il est en particulier possible d'introduire un processus auxiliaire $U_1^N = (U_1, \dots, U_N)$ à valeurs dans un espace fini et pour que le calcul rapide de $p(r_{n+1}|y_1^{n+1})$ soit possible, il est alors suffisant de supposer la markovianité du triplet (R_1^N, U_1^N, Y_1^N) . De tels modèles, dans lesquels (R_1^N, U_1^N) n'est pas nécessairement markovien, s'avèrent très

efficaces en situations non stationnaires (Lanchantin *et al.*, 2011 ; Boudaren *et al.*, 2012a), ce qui est à l'origine de la présente étude. En effet, ainsi que nous allons le montrer, il est possible d'étendre la famille MCCLSM des triplets $T_1^N = (X_1^N, R_1^N, Y_1^N)$ en étendant (R_1^N, Y_1^N) , dans différentes directions, par l'introduction d'un processus auxiliaire $U_1^N = (U_1, \dots, U_N)$ et en considérant la markovianité du triplet (R_1^N, U_1^N, Y_1^N) , ce qui mène à un quadruplet markovien $(X_1^N, R_1^N, U_1^N, Y_1^N)$. Notons qu'outre les différentes stationnarités mentionnées plus haut, U_1^N peut modéliser différentes autres situations mentionnées à la Section 4.

L'objet de cet article est donc de présenter un modèle général $T_1^N = (X_1^N, R_1^N, U_1^N, Y_1^N)$ étendant la famille MCCLSM et de montrer, dans un cas particulier, son intérêt en situation de sauts non stationnaires. L'organisation de l'article est la suivante. Dans la Section suivante, nous rappelons un résultat récent modifiant le modèle classique (1.1)-(1.3) et aboutissant au modèle général appelé « Modèle couple conditionnellement Gaussien ». Dans la Section 3 nous rappelons comment il est possible de définir un modèle proche du modèle général mais permettant un filtrage rapide, car appartenant à la famille des « modèles cachés conditionnellement linéaires à sauts ». La Section 4 introduit le « modèle caché conditionnellement linéaire à sauts non stationnaires », par l'ajout d'un processus auxiliaire. Enfin, la Section 5 procure quelques résultats numériques d'expériences qui mettent en évidence l'intérêt de cette nouvelle modélisation pour le filtrage non-stationnaire en ligne de séries temporelles, et la section 6 contient les conclusions et des perspectives.

2. Généralisation du modèle CGLSSM

Considérons le modèle CGLSSM (1.1)-(1.3) et écrivons (1.2)-(1.3) matriciellement:

$$R_1^N \text{ est une chaîne de Markov ;} \quad (2.1)$$

$$\begin{bmatrix} X_{n+1} \\ Y_{n+1} \end{bmatrix} = \begin{bmatrix} A_{n+1}(R_{n+1}) & 0 \\ B_{n+1}(R_{n+1})A_{n+1}(R_{n+1}) & 0 \end{bmatrix} \begin{bmatrix} X_n \\ Y_n \end{bmatrix} + \begin{bmatrix} C_{n+1}(R_{n+1}) & 0 \\ B_{n+1}(R_{n+1})C_{n+1}(R_{n+1}) & D_{n+1}(R_{n+1}) \end{bmatrix} \begin{bmatrix} U_{n+1} \\ V_{n+1} \end{bmatrix}, \quad (2.2)$$

Il est possible de généraliser (2.1)-(2.2) au modèle suivant (on pose $R_n^{n+1} = (R_n, R_{n+1})$), appelé « Modèle couple conditionnellement gaussien » (en anglais « Conditionally Gaussian Pairwise Markov Model »), qui sera noté CGPMM (Abbassi *et al.*, 2011):

$$R_1^N \text{ est une chaîne de Markov ;} \quad (2.3)$$

$$\begin{bmatrix} X_{n+1} \\ Y_{n+1} \end{bmatrix} = \begin{bmatrix} A_{n+1}^1(R_n^{n+1}) & A_{n+1}^2(R_n^{n+1}) \\ A_{n+1}^3(R_n^{n+1}) & A_{n+1}^4(R_n^{n+1}) \end{bmatrix} \begin{bmatrix} X_n \\ Y_n \end{bmatrix} + \begin{bmatrix} B_{n+1}^1(R_n^{n+1}) & B_{n+1}^2(R_n^{n+1}) \\ B_{n+1}^3(R_n^{n+1}) & B_{n+1}^4(R_n^{n+1}) \end{bmatrix} \begin{bmatrix} U_{n+1} \\ V_{n+1} \end{bmatrix}, \quad (2.4)$$

avec la loi de (X_1, Y_1) et les lois $p(x_1, y_1 | r_1)$, supposées gaussiennes, données, $A_{n+1}^i(R_n^{n+1})$, $B_{n+1}^i(R_n^{n+1})$ des matrices de tailles appropriées dépendantes des sauts, et $(U_2, V_2), \dots, (U_N, V_N)$ un bruit blanc gaussien de matrice de covariance identité. Notons que nous avons deux types de généralisations. Le premier consiste en considérant des termes quelconques au lieu des termes nuls dans les deux matrices dans (2.2). Le deuxième consiste en extension de la dépendance des termes de ces matrices de R_{n+1} à (R_n, R_{n+1}) .

La plus grande généralité du modèle (2.3)-(2.4) par rapport au modèle (2.1)-(2.2) apparaît ainsi avec évidence ; cependant, il garde les propriétés analogues à celles de ce dernier et le même type de méthodes approchées peut être utilisé. En particulier, on peut aisément étendre les filtres particuliers utilisés dans le modèle classique au CGPMM (2.3)-(2.4) (Abbassi *et al.*, 2011). Comme dans les cas classiques, il est possible, dans un CGPMM, d'utiliser le filtre de Kalman à R_1^N connu. En effet, conditionnellement à R_1^N , le couple (X_1^N, Y_1^N) est markovien (sans que X_1^N le soit nécessairement) et il est alors possible d'utiliser la méthode généralisant le filtre de Kalman classique (Pieczynski et Desbouvries, 2003), que nous reformulons ci-après de manière quelque peu modifiée.

En oubliant la dépendance de R_1^N , nous avons le système suivant :

$$\begin{bmatrix} X_{n+1} \\ Y_{n+1} \end{bmatrix} = \begin{bmatrix} A_{n+1}^1 & A_{n+1}^2 \\ A_{n+1}^3 & A_{n+1}^4 \end{bmatrix} \begin{bmatrix} X_n \\ Y_n \end{bmatrix} + \begin{bmatrix} B_{n+1}^1 & B_{n+1}^2 \\ B_{n+1}^3 & B_{n+1}^4 \end{bmatrix} \begin{bmatrix} U_{n+1} \\ V_{n+1} \end{bmatrix} \quad (2.5)$$

Le problème est donc de calculer $p(x_{n+1}|y_1^{n+1})$ à partir de $p(x_n|y_1^n)$, ce qui est fait de la manière suivante. Notons, pour simplifier, $p(x_n|y_1^n) = N(\mu_n, \Sigma_n)$, $p(x_{n+1}|y_1^{n+1}) = N(\mu_{n+1}, \Sigma_{n+1})$, $B_{n+1} = \begin{bmatrix} B_{n+1}^1 & B_{n+1}^2 \\ B_{n+1}^3 & B_{n+1}^4 \end{bmatrix}$ et $W_{n+1} = \begin{bmatrix} U_{n+1} \\ V_{n+1} \end{bmatrix}$. Pour y_1^n fixé, (2.5) montre que les variables X_n , X_{n+1} , et Y_{n+1} vérifient $\begin{bmatrix} X_{n+1} \\ Y_{n+1} \end{bmatrix} = \begin{bmatrix} A_{n+1}^1 X_n + A_{n+1}^2 y_n \\ A_{n+1}^3 X_n + A_{n+1}^4 y_n \end{bmatrix} + B_{n+1} W_{n+1} = \begin{bmatrix} A_{n+1}^1 \\ A_{n+1}^3 \end{bmatrix} X_n + \begin{bmatrix} A_{n+1}^2 \\ A_{n+1}^4 \end{bmatrix} y_n + B_{n+1} W_{n+1}$, et donc que $p(x_{n+1}, y_{n+1}|y_1^n)$ est une loi gaussienne avec moyenne et variance données par

$$\begin{bmatrix} m_{n+1}^X \\ m_{n+1}^Y \end{bmatrix} = \begin{bmatrix} A_{n+1}^1 \\ A_{n+1}^3 \end{bmatrix} \mu_n + \begin{bmatrix} A_{n+1}^2 \\ A_{n+1}^4 \end{bmatrix} y_n; \quad (2.6)$$

$$\begin{bmatrix} S_{n+1}^{XX} & S_{n+1}^{XY} \\ S_{n+1}^{YX} & S_{n+1}^{YY} \end{bmatrix} = \begin{bmatrix} A_{n+1}^1 \\ A_{n+1}^3 \end{bmatrix} \Sigma_n \begin{bmatrix} A_{n+1}^1 & A_{n+1}^3 \end{bmatrix}^T + B_{n+1} B_{n+1}^T \quad (2.7)$$

Alors $p(x_{n+1}|y_1^{n+1}) = p(x_{n+1}|y_{n+1}, y_1^n)$ est une loi gaussienne – obtenue classiquement à partir de la loi gaussienne $p(x_{n+1}, y_{n+1}|y_1^n)$, avec moyenne et variance données par

$$\mu_{n+1} = S_{n+1}^{XY} (S_{n+1}^{YY})^{-1} (y_{n+1} - m_{n+1}^Y) + m_{n+1}^X; \quad (2.8)$$

$$\Sigma_{n+1} = S_{n+1}^{XX} - S_{n+1}^{XY} (S_{n+1}^{YY})^{-1} S_{n+1}^{YX}. \quad (2.9)$$

Ainsi le filtre de Kalman fonctionne dans le modèle (2.5), et coïncide avec le filtre de Kalman classique lorsque (2.3)-(2.4) est réduit à (2.1)-(2.2) (à $R_1^N = r_1^N$ connu). Finalement, comme dans le CGLSSM classique, le filtre de Kalman peut être utilisé dans le CGPMM lorsque $R_1^N = r_1^N$ est connu. Bien entendu, lorsque R_1^N n'est pas connu, la mise en place du filtre optimal dans CGPMM présente les mêmes difficultés calculatoires que dans CGLSSM ; cependant, ces difficultés n'étant pas plus complexes, les techniques de type MCMCS utilisées dans CGLSSM s'étendent aisément aux CGPMM (Abbassi *et al.*, 2011).

3. Filtrage rapide exact dans un cas particulier de CGPMM

Considérons le cas particulier du CGPMM (2.3)-(2.4) suivant:

$$R_1^N \text{ est une chaîne de Markov}; \quad (3.1)$$

$$\begin{bmatrix} X_{n+1} \\ Y_{n+1} \end{bmatrix} = \begin{bmatrix} A_{n+1}^1(R_n^{n+1}) & A_{n+1}^2(R_n^{n+1}) \\ 0 & A_{n+1}^4(R_n^{n+1}) \end{bmatrix} \begin{bmatrix} X_n \\ Y_n \end{bmatrix} + \begin{bmatrix} B_{n+1}^1(R_n^{n+1}) & B_{n+1}^2(R_n^{n+1}) \\ B_{n+1}^3(R_n^{n+1}) & B_{n+1}^4(R_n^{n+1}) \end{bmatrix} \begin{bmatrix} U_{n+1} \\ V_{n+1} \end{bmatrix}, \quad (3.2)$$

ainsi la particularité consiste à considérer $A_{n+1}^3(R_n^{n+1}) = 0$. Montrons, en reprenant les idées de (Abbassi *et al.*, 2011), que le filtrage exact rapide est possible, ce qui signifie que la complexité est linéaire en n , et précisons le déroulement du filtre. Cette possibilité vient du fait que le modèle (3.1)-(3.2) fait partie de la famille des modèles dits « Modèle caché conditionnellement linéaire à sauts », en anglais « conditionally Markov switching hidden

linear models », qui seront notés CMSHLM dans la suite. En effet, ces derniers, introduits dans (Pieczynski, 2011a), autorisent le filtrage exact rapide. Plus précisément, un CMSHLM est défini par :

$$T_1^N = (X_1^N, R_1^N, Y_1^N) \text{ est Markov avec } p(r_{n+1}, y_{n+1} | x_n, r_n, y_n) = p(r_{n+1}, y_{n+1} | r_n, y_n) \\ \text{(le couple } (R_1^N, Y_1^N) \text{ est alors Markov);} \quad (3.3)$$

$$X_{n+1} = F_{n+1}(R_n^{n+1}, Y_n^{n+1})X_n + G_{n+1}(R_n^{n+1}, Y_n^{n+1})W_{n+1} + H_{n+1}(R_n^{n+1}, Y_n^{n+1}), \quad (3.4)$$

avec $F_{n+1}(R_n^{n+1}, Y_n^{n+1})$, $G_{n+1}(R_n^{n+1}, Y_n^{n+1})$ des matrices de dimensions appropriées, W_{n+1} un bruit blanc, et $H_{n+1}(R_n^{n+1}, Y_n^{n+1})$ des vecteurs aléatoires de dimension appropriée.

$p(r_{n+1} | y_1^{n+1})$ et $E[X_{n+1} | r_{n+1}, y_1^{n+1}]$ peuvent alors être calculés à partir de $p(r_{n+1}, y_{n+1} | r_n, y_n)$, $F_{n+1}(r_n^{n+1}, y_n^{n+1})$, $H_{n+1}(r_n^{n+1}, y_n^{n+1})$, $p(r_n | y_1^n)$, et $E[X_n | r_n, y_1^n]$ avec une complexité indépendante de n de la manière suivante :

$$p(r_{n+1} | y_1^{n+1}) = \frac{\sum_{r_n} p(r_{n+1}, y_{n+1} | r_n, y_n) p(r_n | y_1^n)}{\sum_{r_n, r_{n+1}} p(r_{n+1}, y_{n+1} | r_n, y_n) p(r_n | y_1^n)}, \quad (3.5)$$

$$E[X_{n+1} | r_{n+1}, y_1^{n+1}] = \sum_{r_n} p(r_n | r_{n+1}, y_1^{n+1}) [F_{n+1}(r_n^{n+1}, y_n^{n+1}) E[X_n | r_n, y_1^n] + H_{n+1}(r_n^{n+1}, y_n^{n+1})], \quad (3.6)$$

avec

$$p(r_n | r_{n+1}, y_1^{n+1}) = \frac{p(r_{n+1}, y_{n+1} | r_n, y_n) p(r_n | y_1^n)}{\sum_{r_n} p(r_{n+1}, y_{n+1} | r_n, y_n) p(r_n | y_1^n)}. \quad (3.7)$$

Pour la mise en œuvre pratique du filtre à partir de (3.1)-(3.2) il nous faut préciser les liens entre les coefficients des matrices de (3.2) et les quantités $F_{n+1}(R_n^{n+1}, Y_n^{n+1})$, $G_{n+1}(R_n^{n+1}, Y_n^{n+1})$, et $H_{n+1}(R_n^{n+1}, Y_n^{n+1})$ de (3.4). En oubliant, pour simplifier, la dépendance de n et R_n^{n+1} de la loi de $(X_{n+1}, Y_{n+1})^T$ conditionnelle à $(X_n, Y_n)^T = (x_n, y_n)^T$, cette dernière est, selon (3.2), la loi gaussienne de moyenne

$$\begin{bmatrix} m_1 \\ m_2 \end{bmatrix} = \begin{bmatrix} A_{n+1}^1(r_n^{n+1})x_n + A_{n+1}^2(r_n^{n+1})y_n \\ A_{n+1}^4(r_n^{n+1})y_n \end{bmatrix}, \quad (3.8)$$

et de matrice de variance-covariance

$$\begin{bmatrix} \gamma_1 & \gamma_2 \\ \gamma_3 & \gamma_4 \end{bmatrix} = \begin{bmatrix} B_{n+1}^1(R_n^{n+1}) & B_{n+1}^2(R_n^{n+1}) \\ B_{n+1}^3(R_n^{n+1}) & B_{n+1}^4(R_n^{n+1}) \end{bmatrix} \begin{bmatrix} B_{n+1}^1(R_n^{n+1}) & B_{n+1}^2(R_n^{n+1}) \\ B_{n+1}^3(R_n^{n+1}) & B_{n+1}^4(R_n^{n+1}) \end{bmatrix}^T. \quad (3.9)$$

La loi de X_{n+1} conditionnelle à $Y_{n+1} = y_{n+1}$ (qui est donc celle de X_{n+1} conditionnelle à $Y_{n+1} = y_{n+1}$, $Y_n = y_n$, et $X_n = x_n$) est alors classiquement la loi gaussienne de moyenne

$$m_1 + \gamma_2(\gamma_4)^{-1}(y_{n+1} - m_2), \quad (3.10)$$

et de variance

$$\gamma_1 - \gamma_2(\gamma_4)^{-1}\gamma_3. \quad (3.11)$$

En considérant γ^* la matrice telle que

$$\gamma^*(\gamma^*)^T = \gamma_1 - \gamma_2(\gamma_4)^{-1}\gamma_3, \quad (3.12)$$

la variable X_{n+1} peut s'écrire

$$X_{n+1} = m_1 + \gamma_2(\gamma_4)^{-1}(Y_{n+1} - m_2) + \gamma^*W_{n+1}, \quad (3.13)$$

avec W_{n+1} un bruit blanc. En reportant (3.8) et (3.9) dans (3.10)-(3.13) on trouve

$$\begin{aligned} X_{n+1} &= A_{n+1}^1(r_n^{n+1})X_n + A_{n+1}^2(r_n^{n+1})Y_n + \gamma_2(\gamma_4)^{-1}(Y_{n+1} - A_{n+1}^4(r_n^{n+1})Y_n) + \gamma^*W_{n+1} = \\ &A_{n+1}^1(r_n^{n+1})X_n + [A_{n+1}^2(r_n^{n+1}) - \gamma_2(\gamma_4)^{-1}A_{n+1}^4(r_n^{n+1})]Y_n + \gamma_2(\gamma_4)^{-1}Y_{n+1} + \gamma^*W_{n+1}, \end{aligned}$$

et donc finalement

$$F_{n+1}(R_n^{n+1}, Y_n^{n+1}) = A_{n+1}^1(r_n^{n+1}); \quad (3.14)$$

$$G_{n+1}(R_n^{n+1}, Y_n^{n+1}) = \gamma^*; \quad (3.15)$$

$$H_{n+1}(R_n^{n+1}, Y_n^{n+1}) = [A_{n+1}^2(R_n^{n+1}) - \gamma_2(\gamma_4)^{-1}A_{n+1}^4(R_n^{n+1})]Y_n + \gamma_2(\gamma_4)^{-1}Y_{n+1}, \quad (3.16)$$

avec γ^* donné par (3.9) et (3.12), et γ_2, γ_4 donnés par (3.9).

Remarque

Sachant que les calculs rapides du filtre optimal sont possibles dans un CMSHLM et ne sont pas possibles dans un CGLSSM classique, une question qui se pose naturellement est de savoir dans quelle mesure ces modèles sont différents. Cette question a été étudiée de plusieurs manières différentes :

(i) considérons l'égalité (2.2) sous forme compacte $Z_{n+1} = A_{n+1}(R_{n+1})Z_n + B_{n+1}(R_{n+1})W_{n+1}$. Les matrices de covariance $\Gamma_{n+1}(R_1^{n+1})$, $\Gamma_n(R_1^n)$ de Z_{n+1} et Z_n (conditionnelles à R_1^{n+1}) sont alors classiquement liées par la relation $\Gamma_{n+1}(R_1^{n+1}) = A_{n+1}(R_{n+1})\Gamma_n(R_1^n)A_{n+1}^T(R_{n+1}) + B_{n+1}(R_{n+1})B_{n+1}^T(R_{n+1})$. Il est alors possible de considérer $Z_{n+1} = A_{n+1}^*(R_{n+1})Z_n + B_{n+1}^*(R_{n+1})W_{n+1}$ de forme (3.2), avec A_{n+1}^* et $B_{n+1}^*(R_{n+1})$ telles que pour $\Gamma_n(R_1^n) = \Gamma_n^*(R_1^n)$ on ait $\Gamma_{n+1}^*(R_1^{n+1}) = A_{n+1}^*(R_{n+1})\Gamma_n(R_1^n)A_{n+1}^{*T}(R_{n+1}) + B_{n+1}^*(R_{n+1})B_{n+1}^{*T}(R_{n+1})$. De proche en proche on a donc un modèle (3.2) présentant les mêmes matrices de covariances que le modèle (2.2) (Pieczynski *et al.*, 2013b).

(ii) Dans le modèle classique (2.2) les lois marginales $p(x_n, y_n)$ sont des mélanges dont le nombre des composantes croît exponentiellement avec le nombre d'observations. Cette propriété implique que deux observateurs utilisant deux modèles (2.2) avec les mêmes transitions, mais ayant commencé les observations à des instants différents, ne peuvent pas avoir les mêmes lois $p(x_n, y_n)$, c'est qui est manifestement un inconvénient. Il est alors possible de modifier les matrices $C_{n+1}(R_{n+1})$ dans (1.2) - ou dans (2.2) - en les faisant dépendre de R_n^{n+1} de façon à ce que les lois marginales $p(x_n, y_n)$ soient des mélanges dont le nombre des composantes est constant. A chaque élément de cette famille des modèles classiques « modifiés » il est alors possible d'associer un modèle (3.1)-(3.2) très proche : pour tout $n = 1, \dots, N-1$, les lois $p(x_n | r_n)$, $p(y_{n+1} | r_{n+1})$, $p(x_n, x_{n+1} | r_n, r_{n+1})$, $p(x_{n+1}, y_n | r_n, r_{n+1})$, $p(y_n, y_{n+1} | r_n, r_{n+1})$, $p(x_n, y_n | r_n)$, et $p(x_{n+1}, y_{n+1} | r_{n+1})$ sont les mêmes dans les deux modèles et la seule différence réside dans le fait que les lois $p(x_n, y_{n+1} | r_n, r_{n+1})$ sont différentes. En particulier, $p(x_{n+1} | x_n, r_n, r_{n+1})$ et $p(y_n | x_n, r_n)$, que l'on utilise pour définir les modèles classiques, sont les mêmes (Derrode et Pieczynski, 2013).

(iii) De manière générale, le modèle classique CGLSSM est déterminé, à sauts connus, par la loi $p(x_1, y_1)$ de (X_1, Y_1) et deux suites de lois : $p(x_{n+1} | x_n)$, et $p(y_n | x_n)$. On peut alors montrer qu'il existe un modèle (3.1)-(3.2) dans lequel ces lois, à sauts donnés, sont les mêmes (Petetin et Desbouvries, 2013 ; Petetin, 2013). On est devant

un argument important en faveur du modèle (3.1)-(3.2) car, dans la pratique, la loi $p(x_1, y_1)$ et les deux suites de lois $p(x_{n+1}|x_n)$, $p(y_n|x_n)$ sont celles qui modélisent les propriétés « physiques » du phénomène à étudier.

4. Filtrage rapide exact dans les « Modèles cachés conditionnellement linéaires à sauts marginalement markoviens » (MCCLSM).

Considérons un processus $T_1^N = (X_1^N, R_1^N, U_1^N, Y_1^N)$, avec X_1^N continu caché, Y_1^N continu observé, R_1^N fini modélisant les sauts, et U_1^N processus auxiliaire fini $U_n \in \Lambda = \{1, \dots, L\}$ pouvant admettre différentes interprétations.

Définition 4.1

$T_1^N = (X_1^N, R_1^N, U_1^N, Y_1^N)$ est un « modèle caché conditionnellement linéaire à sauts marginalement markoviens » (MCCLSM) à sauts si pour tout $n = 1, \dots, N-1$:

$$T_1^N = (X_1^N, R_1^N, U_1^N, Y_1^N) \text{ est de Markov;} \quad (4.1)$$

$$p(r_{n+1}, u_{n+1}, y_{n+1} | x_n, r_n, u_n, y_n) = p(r_{n+1}, u_{n+1}, y_{n+1} | r_n, u_n, y_n); \quad (4.2)$$

$$X_{n+1} = F_{n+1}(R_n^{n+1}, U_n^{n+1}, Y_n^{n+1})X_n + G_{n+1}(R_n^{n+1}, U_n^{n+1}, Y_n^{n+1})W_{n+1} + H_{n+1}(R_n^{n+1}, U_n^{n+1}, Y_n^{n+1}). \quad (4.3)$$

avec $F_{n+1}(R_n^{n+1}, U_n^{n+1}, Y_n^{n+1})$, $G_{n+1}(R_n^{n+1}, U_n^{n+1}, Y_n^{n+1})$ des matrices de dimensions appropriées, W_{n+1} un bruit blanc, et $H_{n+1}(R_n^{n+1}, U_n^{n+1}, Y_n^{n+1})$ des vecteurs aléatoires de dimension appropriée.

Proposition 4.1

Soit $T_1^N = (X_1^N, R_1^N, U_1^N, Y_1^N)$ un MCCLSM. Alors $p(r_{n+1}, u_{n+1} | y_1^{n+1})$, $E[X_{n+1} | r_{n+1}, u_{n+1}, y_1^{n+1}]$, et $E[X_{n+1} X_{n+1}^T | r_{n+1}, u_{n+1}, y_1^{n+1}]$, qui donnent $E[X_{n+1} | y_1^{n+1}]$, et $Var[X_{n+1} | y_1^{n+1}]$, peuvent être calculés à partir de $p(r_{n+1}, u_{n+1}, y_{n+1} | r_n, u_n, y_n)$, $F_{n+1}(R_n^{n+1}, U_n^{n+1}, Y_n^{n+1})$, $G_{n+1}(R_n^{n+1}, U_n^{n+1}, Y_n^{n+1})$, $H_{n+1}(R_n^{n+1}, U_n^{n+1}, Y_n^{n+1})$, $p(r_n, u_n | y_1^n)$, $E[X_n | r_n, u_n, y_1^n]$, et $E[X_n X_n^T | r_n, u_n, y_1^n]$ avec une complexité indépendante de n . En posant $V_1^N = (R_1^N, U_1^N)$, nous avons :

$$p(v_{n+1} | y_1^{n+1}) = \frac{\sum_{v_n} p(v_{n+1}, y_{n+1} | v_n, y_n) p(v_n | y_1^n)}{\sum_{v_n, v_{n+1}} p(v_{n+1}, y_{n+1} | v_n, y_n) p(v_n | y_1^n)}; \quad (4.4)$$

$$E[X_{n+1} | v_{n+1}, y_1^{n+1}] = \sum_{v_n} p(v_n | v_{n+1}, y_1^{n+1}) [F_{n+1}(v_n^{n+1}, y_n^{n+1}) E[X_n | v_n, y_1^n] + H_{n+1}(v_n^{n+1}, y_n^{n+1})]; \quad (4.5)$$

avec

$$p(v_n | v_{n+1}, y_1^{n+1}) = \frac{p(v_{n+1}, y_{n+1} | v_n, y_n) p(v_n | y_1^n)}{\sum_{v_n} p(v_{n+1}, y_{n+1} | v_n, y_n) p(v_n | y_1^n)}; \quad (4.6)$$

$$E[X_{n+1} X_{n+1}^T | v_{n+1}, y_1^{n+1}] = \sum_{v_n} p(v_n | v_{n+1}, y_1^{n+1}) [A_{n+1}(v_n^{n+1}, y_1^{n+1})] \quad (4.7)$$

avec

$$\begin{aligned} A_{n+1}(v_n^{n+1}, y_1^{n+1}) &= F_{n+1}(v_n^{n+1}, y_n^{n+1}) E[X_n X_n^T | v_n, y_1^n] F_{n+1}^T(v_n^{n+1}, y_n^{n+1}) + \\ &F_{n+1}(v_n^{n+1}, y_n^{n+1}) E[X_n | v_n, y_1^n] H_{n+1}^T(v_n^{n+1}, y_n^{n+1}) + G_{n+1}(v_n^{n+1}, y_n^{n+1}) G_{n+1}^T(v_n^{n+1}, y_n^{n+1}) + \\ &H_{n+1}(v_n^{n+1}, y_n^{n+1}) E[X_n^T | v_n, y_1^n] F_{n+1}^T(v_n^{n+1}, y_n^{n+1}) + H_{n+1}(v_n^{n+1}, y_n^{n+1}) H_{n+1}^T(v_n^{n+1}, y_n^{n+1}) \end{aligned} \quad (4.8)$$

$$p(r_{n+1} | y_1^{n+1}) = \sum_{u_{n+1}} p(v_{n+1} | y_1^{n+1}), \quad p(u_{n+1} | y_1^{n+1}) = \sum_{r_{n+1}} p(v_{n+1} | y_1^{n+1}); \quad (4.9)$$

$$E[X_{n+1} | y_1^{n+1}] = \sum_{v_{n+1}} p(v_{n+1} | y_1^{n+1}) E[X_{n+1} | v_{n+1}, y_1^{n+1}]. \quad (4.10)$$

Démonstration

(4.4) vient de la markovianité du couple (V_1^N, Y_1^N) :

$$\begin{aligned} p(v_{n+1} | y_1^{n+1}) &= \sum_{v_n} p(v_n, v_{n+1} | y_1^{n+1}) = \sum_{v_n} p(v_n, v_{n+1} | y_1^{n+1}) = \sum_{v_n} \frac{p(v_n, v_{n+1}, y_{n+1} | y_1^n)}{p(y_{n+1} | y_1^n)} = \\ &\frac{\sum_{v_n} p(v_{n+1}, y_{n+1} | v_n, y_n) p(v_n | y_1^n)}{\sum_{v_n, v_{n+1}} p(v_{n+1}, y_{n+1} | v_n, y_n) p(v_n | y_1^n)}; \end{aligned}$$

Pour montrer (4.5) on considère $E[X_{n+1} | v_{n+1}, y_1^{n+1}]$ avec X_{n+1} donnée par (4.3) :

$$\begin{aligned} E[X_{n+1} | v_{n+1}, y_1^{n+1}] &= E[F_{n+1}(V_n^{n+1}, Y_n^{n+1}) X_n + G_{n+1}(V_n^{n+1}, Y_n^{n+1}) W_{n+1} + H_{n+1}(V_n^{n+1}, Y_n^{n+1}) | v_{n+1}, y_1^{n+1}] = \\ &\sum_{v_n} p(v_n | v_{n+1}, y_1^{n+1}) E[F_{n+1}(V_n^{n+1}, Y_n^{n+1}) X_n + G_{n+1}(V_n^{n+1}, Y_n^{n+1}) W_{n+1} + H_{n+1}(V_n^{n+1}, Y_n^{n+1}) | v_n, v_{n+1}, y_1^{n+1}] = \\ &\sum_{v_n} p(v_n | v_{n+1}, y_1^{n+1}) [F_{n+1}(v_n^{n+1}, y_n^{n+1}) E[X_n | v_n, v_{n+1}, y_1^{n+1}] + H_{n+1}(v_n^{n+1}, y_n^{n+1})] \end{aligned}$$

Selon l'hypothèse (4.2) X_n et (V_{n+1}, Y_{n+1}) sont indépendants conditionnellement à (V_n, Y_n) , ce qui implique

$$E[X_n | v_n, v_{n+1}, y_1^{n+1}] = E[X_n | v_n, y_1^n], \text{ d'où (4.5) ;}$$

$$(4.6) \text{ vient de } p(v_n, v_{n+1} | y_1^{n+1}) = \frac{p(v_n, v_{n+1}, y_{n+1} | y_1^n)}{p(y_{n+1} | y_1^n)} = \frac{p(v_{n+1}, y_{n+1} | v_n, y_n) p(v_n | y_1^n)}{p(y_{n+1} | y_1^n)};$$

La démonstration de (4.7)-(4.8) est analogue à celle de (4.5) ; (4.9) et (4.10) sont immédiats.

Remarque

Malgré la similitude mathématique entre les modèles MCCLSM et MCCLSM, la différence, apparaissant au niveau de l'interprétation des processus cachés, peut être très importante du point de vue pratique. Dans le problème de segmentation, utiliser une chaîne de Markov triplet (R_1^N, U_1^N, Y_1^N) , où U_1^N modélise différentes stationnarités, à la place d'une chaîne de Markov cachée classique (R_1^N, Y_1^N) peut conduire à une amélioration significative des résultats obtenus (Lanchantin *et al.*, 2011). Au plan plus théorique, les différentes

interprétations de U_1^N peuvent mener à différentes modélisations comme les semi-Markov cachés (Lapuyade-Lahorgue et Pieczynski, 2011) ou les chaînes cachées évidentielles (Pieczynski, 2007 ; Boudaren *et al.*, 2012b ; Ramasso et Denoeux, 2013). Notons également l'utilisation récente du processus auxiliaire pour modéliser, dans le cas multivarié, les états à retard existant entre les canaux observés (Le Cam, 2013). Toutes ces modélisations peuvent être utilisées dans le cadre d'un MCCLSM introduit dans cet article, ce qui constitue autant de perspectives pour les travaux futurs.

5. Expérimentations

Nous présentons dans cette section les résultats de simulations dans lesquelles les processus X_1^N et Y_1^N sont scalaires (*i.e.* $m = q = 1$), avec deux classes pour R_1^N ($K = 2$) et trois classes pour U_1^N ($L = 3$). La longueur des chaînes est fixée à $N = 2000$ échantillons, et les résultats sont des moyennes de 300 expériences indépendantes. Nous considérerons uniquement des systèmes homogènes, ce qui signifie que les paramètres $A_{n+1}^i(V_n^{n+1})$, $i = 1, 2, 4$, et $B_{n+1}^i(V_n^{n+1})$, $i = 1, \dots, 4$ dans (3.2) - avec V_n^{n+1} à la place de R_n^{n+1} - ne dépendent pas de n , et tels que le couple $V_1^N = (R_1^N, U_1^N)$ forme une chaîne de Markov homogène définie de la manière suivante :

- U_1^N est une chaîne de Markov homogène définie par la distribution de U_1 , donnée par $\left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$, et les matrices de transition données par

$$M_U = \begin{pmatrix} 0.95 & 0.05 & 0 \\ 0 & 0.95 & 0.05 \\ 0.05 & 0 & 0.95 \end{pmatrix}, \quad (5.1)$$

- R_1^N est une chaîne de Markov conditionnellement à U_1^N , définie par la distribution de R_1 donnée par $\left(\frac{1}{2}, \frac{1}{2}\right)$, et les matrices de transition donnant les lois de R_{n+1} sachant R_n (et $U_{n+1} = 1, 2, 3$) données par

$$M_R^1 = \begin{pmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{pmatrix}, M_R^2 = \begin{pmatrix} 0.65 & 0.35 \\ 0.35 & 0.65 \end{pmatrix}, M_R^3 = \begin{pmatrix} 0.35 & 0.65 \\ 0.65 & 0.35 \end{pmatrix}. \quad (5.2)$$

La loi de $V_1^N = (R_1^N, U_1^N)$ est donc définie par $p(v_1)$ sur $\{1, 2\} \times \{1, 2, 3\}$, donnée par $\left(\frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}\right)$, et par les transitions $p(v_2|v_1) = p(r_2, u_2|r_1, u_1) = p(u_2|u_1)p(r_2|r_1, u_2)$, où $p(u_2|u_1)$ est donnée par (5.1), et les lois $p(r_2|r_1, u_2)$ sont données, pour $u_2 = 1, 2, 3$, par (5.2). Notons que nous considérons le cas de particulier de $p(v_2|v_1) = p(u_2|r_1, u_1)p(r_2|r_1, u_1, u_2)$, en supposant $p(u_2|r_1, u_1) = p(u_2|u_1)$ et $p(r_2|r_1, u_1, u_2) = p(r_2|r_1, u_2)$. Cependant, même ce cas assez particulier produit un modèle dans lequel R_1^N n'est markovien, un cas donc différent des modèles classiques où il l'est.

Oublions un instant la dépendance de (X_1^N, Y_1^N) des sauts V_1^N et notons $Z_1^2 = (Z_1, Z_2) = (X_1, Y_1, X_2, Y_2)$. La distribution gaussienne de Z_1^2 (conditionnelle à $(V_1, V_2) = (v_1, v_2)$) est donnée par la matrice de covariance suivante :

$$\Gamma^{Z_1^2} = \begin{matrix} & X_1 & Y_1 & X_2 & Y_2 \\ \begin{matrix} X_1 \\ Y_1 \\ X_2 \\ Y_2 \end{matrix} & \begin{bmatrix} \sigma_X^2 & b & a & d \\ b & \sigma_Y^2 & e & c \\ a & e & \sigma_X^2 & b \\ d & c & b & \sigma_Y^2 \end{bmatrix} & = & \begin{bmatrix} \Gamma & \Sigma^T \\ \Sigma & \Gamma \end{bmatrix} \end{matrix}, \quad (5.3)$$

avec $\Gamma = \begin{bmatrix} \sigma_X^2 & b \\ b & \sigma_Y^2 \end{bmatrix}$ and $\Sigma = \begin{bmatrix} a & e \\ d & c \end{bmatrix}$. Ainsi la distribution of Z_1^2 est définie par 2 variances σ_X^2, σ_Y^2 et 5 co-variances a, b, c, d et e , telles que Γ^{-1} soit une matrice définie positive. Le graphe de dépendance est présenté en Figure 1.

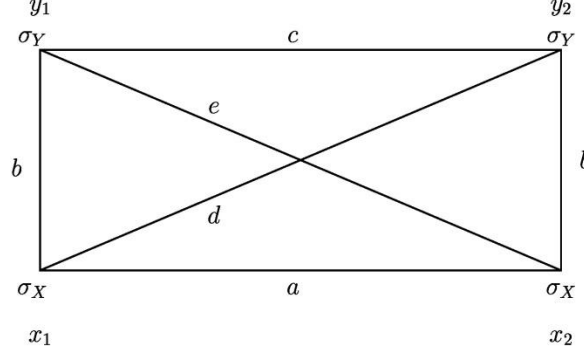


Figure 1 : Graphe de dépendance de $p(x_1, y_1, x_2, y_2 | v_1, v_2)$.

Le modèle couple conditionnellement gaussien s'écrit alors

$$\begin{bmatrix} X_{n+1} \\ Y_{n+1} \end{bmatrix} = \begin{bmatrix} A^1 & A^2 \\ A^3 & A^4 \end{bmatrix} \begin{bmatrix} X_n \\ Y_n \end{bmatrix} + \begin{bmatrix} B^1 & B^2 \\ B^3 & B^4 \end{bmatrix} \begin{bmatrix} U_{n+1} \\ V_{n+1} \end{bmatrix}, \quad (5.4)$$

avec

$$\begin{bmatrix} A^1 & A^2 \\ A^3 & A^4 \end{bmatrix} = \begin{bmatrix} a & e \\ d & c \end{bmatrix} \begin{bmatrix} \sigma_X^2 & b \\ b & \sigma_Y^2 \end{bmatrix}^{-1} = \frac{1}{\sigma_X^2 \sigma_Y^2 - b^2} \begin{bmatrix} a\sigma_Y^2 - eb & e\sigma_X^2 - ab \\ d\sigma_Y^2 - cb & c\sigma_X^2 - db \end{bmatrix}, \quad (5.5)$$

et

$$\begin{bmatrix} B^1 & B^2 \\ B^3 & B^4 \end{bmatrix} \begin{bmatrix} B^1 & B^2 \\ B^3 & B^4 \end{bmatrix}^T = \Gamma - \Sigma \Gamma^{-1} \Sigma^T \\ = \begin{bmatrix} \sigma_X^2 & b \\ b & \sigma_Y^2 \end{bmatrix} - \frac{1}{\sigma_X^2 \sigma_Y^2 - b^2} \begin{bmatrix} a\sigma_Y^2 - eb & e\sigma_X^2 - ab \\ d\sigma_Y^2 - cb & c\sigma_X^2 - db \end{bmatrix} \begin{bmatrix} a & d \\ e & c \end{bmatrix}. \quad (5.6)$$

Nous obtenons $A^3 = 0$ lorsque $d = \frac{cb}{\sigma_Y^2}$.

La dépendance de la matrice (5.3) de $V_{n+1} = (R_{n+1}, U_{n+1})$, ainsi celle des matrices (5.5) et (5.6) qui en résulte, est donnée dans le Tableau 1.

U_{n+1}	R_{n+1}	μ_Y	σ_X^2	σ_Y^2	a	b	c	d	e
0	0	0.0	0.5	1.5	0.30	0.60	0.30	0.12	0.40
	1	2.0	1.5	0.8	0.50	0.30	0.30	0.1125	0.80
1	0	0.0	0.5	0.8	0.15	0.60	0.18	0.135	0.25
	1	2.0	0.5	1.	0.30	0.30	0.10	0.03	0.40
2	0	0.0	0.5	1	0.20	0.60	0.20	0.12	0.20
	1	2.0	1	2	0.40	0.20	0.0	0.0	0.60

U_{n+1}	R_{n+1}	μ_Y	A^1	A^2	A^4	B^1	B^2	B^3	B^4
0	0	0.0	0.54	0.05	0.20	0.56	0.00	0.92	0.77
	1	2.0	0.14	0.95	0.38	0.82	0.00	0.00	0.83
1	0	0.0	-0.75	0.87	0.22	0.63	0.00	0.87	0.09
	1	2.0	0.44	0.27	0.10	0.51	0.00	0.51	0.85
2	0	0.0	0.57	-0.14	0.20	0.64	0.00	0.87	0.45
	1	2.0	0.35	0.27	0.00	0.84	0.00	0.24	1.39

Tableau 1 : Valeurs des paramètres utilisés dans les expériences de simulation / restauration.

L'expérience consiste à simuler un quadruplet $(X_1^N, R_1^N, U_1^N, Y_1^N) = (x_1^{2000}, r_1^{2000}, u_1^{2000}, y_1^{2000})$ selon le modèle spécifié par les paramètres du Tableau 1 : on simule (X_1, R_1, U_1, Y_1) en simulant $(R_1, U_1) = (r_1, u_1)$ selon la loi uniforme sur $\{1, 2\} \times \{1, 2, 3\}$ et $(X_1, Y_1) = (x_1, y_1)$ selon la loi gaussienne $p(x_1, y_1 | r_1, u_1)$, cette dernière utilisant les paramètres σ_X^2 , σ_Y^2 , et b donnés également par le Tableau 1, ensuite on utilise les transitions $p(v_2 | v_1) = p(u_2 | u_1) p(r_2 | r_1, u_2)$ pour simuler v_{n+1} sachant v_n et on simule (x_{n+1}, y_{n+1}) sachant v_{n+1} et (x_n, y_n) selon la loi gaussienne $p(x_{n+1}, y_{n+1} | v_{n+1}, x_n, y_n)$ utilisant les paramètres donnés par le tableau. Ensuite, on restaure les données selon quatre filtres :

- le filtre de Kalman utilisant les sauts (à sauts connus, noté F1-SC), *i.e.* on restaure x_1^{2000} à partir de $(r_1^{2000}, u_1^{2000}, y_1^{2000})$;
- le filtre à sauts inconnus (noté F1-SI) à partir de l'algorithme décrit dans la section 4, *i.e.* on restaure $(x_1^{2000}, r_1^{2000}, u_1^{2000})$ à partir de y_1^{2000} uniquement ;
- le filtre de Kalman utilisant les sauts (à sauts connus), mais considérant le modèle triplet (X_1^N, R_1^N, Y_1^N) sans processus auxiliaire (en considérant donc R_1^N à la place de V_1^N), les paramètres de restauration pour ce filtre étant définis « en moyennant » les paramètres du modèle, aboutissant à la distribution de la chaîne de Markov R_1^N donnée par $p(R_1, R_2) = \begin{pmatrix} 0.35 & 0.15 \\ 0.15 & 0.35 \end{pmatrix}$, et les autres paramètres présentés dans le Tableau 2 ;
- Le filtre à sauts inconnus (noté F2-SI), qui correspond à la Proposition 4.1 avec R_1^N à la place de V_1^N , avec les paramètres donnés par $p(R_1, R_2) = \begin{pmatrix} 0.35 & 0.15 \\ 0.15 & 0.35 \end{pmatrix}$ et le Tableau 2.

R_{n+1}	μ_Y	σ_X^2	σ_Y^2	a	b	c	d	e
0	0.0	0.5	1.1	0.22	0.60	0.23	0.13	0.28
1	2.0	1.0	1.27	0.40	0.27	0.13	0.03	0.60

R_{n+1}	μ_Y	A^1	A^2	A^4	B^1	B^2	B^3	B^4
0	0.0	0.36	0.06	0.21	0.64	0.00	0.85	0.57
1	2.0	0.29	0.41	0.11	0.80	0.00	0.26	1.09

Tableau 2 : Valeurs des paramètres utilisés pour la restauration selon le filtre F2.

La qualité de restauration de x est mesurée à l'aide de l'erreur quadratique moyenne (EQM), celles de r et de u , dans le cas à sauts inconnus, le sont par le taux d'erreur de classification (τ). Les résultats moyens obtenus sur 300 expériences sont reportés dans le Tableau 3. Un exemple de restauration de x à saut connu et à sauts inconnus est présenté dans la Figure 2.

	EQM	$\tau(r)$	$\tau(u)$
F1-SC	0.346	-	-
F1-SI	0.616	15.4%	43.0%
F2-SC	0.469	-	-
F2-SI	0.682	17.4%	-

Tableau 3 : Résultats de restauration à sauts connus (F1-SC, F2-SC) et à sauts inconnus (F1-SI, F2-SI).

Les résultats présentés, et d'autres résultats obtenus par des simulations analogues, montrent que, lorsque les données suivent un modèle à sauts non stationnaires proposé, l'utilisation du modèle à sauts stationnaires peut dégrader les résultats obtenus avec le vrai modèle de manière non négligeable.

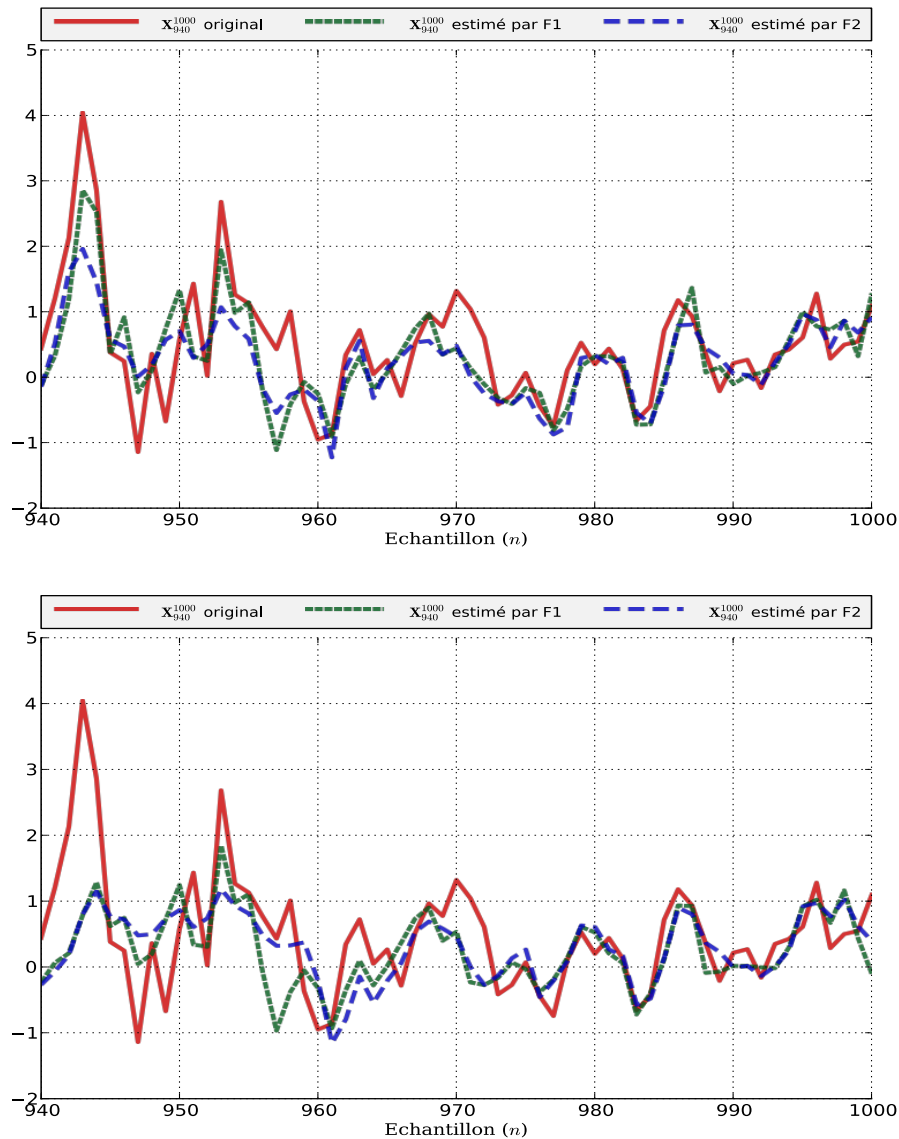


Figure 2. Extraits de données restaurées avec, en haut, les filtres F1 et F2 à sauts connus et, en bas, les filtres F1 et F2 à sauts inconnus.

6. Conclusions et perspectives

Nous avons considéré dans cet article le problème de filtrage statistique optimal, au sens de l'erreur quadratique moyenne, dans des systèmes linéaires à sauts. Notons X le processus caché, Y le processus observé, et R le processus des sauts. Dans les modèles classiques, où les sauts sont modélisés par une chaîne de Markov et le couple (X, Y) est markovien conditionnellement aux sauts, il est impossible de mettre en place un filtrage rapide, dont la complexité serait linéaire en temps, même dans les cas les plus simples où la loi du couple (X, Y) conditionnellement aux sauts est celle d'un système markovien gaussien. Un tel filtrage rapide est possible dans une famille particulière de modèles à sauts, où le couple (X, Y) est markovien conditionnellement aux sauts comme dans les modèles classiques, cependant, le processus caché X ne l'est plus nécessairement (Pieczynski, 2011a). Dans cette famille, que nous appellerons F , le couple (R, Y) est markovien et, comme conséquence, le processus observé Y est markovien conditionnellement aux sauts, ce qui n'est pas le cas dans les modèles classiques. L'apport de cet article est de proposer une extension de cette famille en gardant la possibilité de filtrage optimal rapide. Nous avons introduit un processus discret fini auxiliaire U , et nous avons considéré le quadruplet $T^* = (X, U, R, Y)$. En posant $V = (U, R)$, il est alors suffisant de supposer que le triplet (X, V, Y) appartient à la famille F . Le processus U étant discret, V est également discret, et donc il est possible de faire les mêmes traitements que dans le modèle $T = (X, R, Y)$. T et T^* sont ainsi quasi-identiques au plan mathématique ; cependant, T^* est bien plus général dans le sens suivant. Considérons $T^* = (X, U, R, Y)$ faisant partie de la famille F : (U, R, Y) est alors une chaîne de Markov. Cependant, il est aisé de trouver des lois markoviennes de (U, R, Y) telles que ni R ni (R, Y) ne soient nécessairement markoviens (Lanchantin et al., 2011). Nous pouvons donc avoir des modèles $T^* = (X, U, R, Y)$ dans F sans que $T = (X, R, Y)$, dont la loi est la loi marginale de T^* , ne soit dans F . En particulier, on peut prendre (U, R) markovienne, en obtenant un modèle où la loi des sauts est la loi marginale d'une loi markovienne (les lois semi-markoviennes en font partie). Nous avons présenté des simulations dans le cas très simple où les différentes valeurs de U présentent différentes stationnarités de R , et il s'avère que les résultats obtenus avec U peuvent améliorer de façon significative ceux obtenus sans U , où l'on suppose les sauts stationnaires.

Les perspectives pour ce travail contiennent les différentes extensions en complexifiant la loi de (U, R, Y) ; en particulier, on peut considérer des processus « partiellement » de Markov, autorisant des bruitages à mémoire longue (Lanchantin *et al.*, 2008 ; Pieczynski, 2013a). On pourra alors envisager d'utiliser ces différents modèles à des fins de lissage ou prédictions, en étendant ainsi les premiers résultats obtenus dans le cadre des $T = (X, R, Y)$ dans F , présentés dans (Bardel et Desbouvries, 2012 ; Pieczynski, 2011b). L'estimation des paramètres, en s'inspirant éventuellement de (Fox *et al.*, 2011), constitue une autre perspective intéressante.

Références

- Abbassi, N., Benboudjema, D., and Pieczynski, W. (2011). Kalman filtering approximations in triplet Markov Gaussian switching models, *IEEE Workshop on Statistical Signal Processing (SSP2011)*, Nice, France, 28-30 June.
- Abbassi, N. (2012). Chaînes de Markov triplets et filtrage optimal dans les systèmes à sauts, thèse de doctorat de Telecom SudParis et de l'Université Pierre et Marie Curie (Paris 6), soutenue le 26 avril 2012.
- Al-Anaswah, N. and Wilfling, B. (2011). Identification of speculative bubbles using state-space models with Markov-switching, *Journal of Banking and Finances*, vol. 35, n°5, p. 1073-1086.
- Andrieu, C., Davy, C. M., and Doucet, A. (2003). Efficient particle filtering for jump Markov systems. Application to time-varying autoregressions, *IEEE Trans. on Signal Processing*, vol. 51, n°7, p. 1762-1770.
- Arulampalam, M. S., Maskell, S., Gordon, N. and Clapp, T. (2003). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, *IEEE Trans. on Signal Processing*, vol. 50, n°2, p. 174-188.
- Bardel, N. and Desbouvries, F. (2012). Exact Bayesian prediction in a class of Markov-switching models, *Methodology and Computing in Applied Probability*, vol. 14, n°1, p. 125-134.
- Boudaren, M. Y., Monfrini, E., and Pieczynski, W. (2012a). Unsupervised segmentation of random discrete data hidden with switching noise distributions, *IEEE Signal Processing Letters*, vol. 19, n° 10, p. 619-622.
- Boudaren, M. Y., Pieczynski, W., Monfrini, E., and Aissani, A. (2012b). Dempster-Shafer fusion of multisensor signals in nonstationary Markovian context, *EURASIP Journal on Advances in Signal Processing*, n°134.
- Cappé, O., Moulines, E., and Ryden, T. (2005). Inference in hidden Markov models, Springer.

- Carter, C. and Kohn, R. (1996). Markov chain Monte Carlo in conditionally Gaussian state space models, *Biometrika*, vol. 83, n°3, p. 589-601.
- Carvalho, C. M. and Lopez, H. F. (2007). Simulation-based sequential analysis of Markov switching stochastic volatility models, *Computational Statistics & Data Analysis*, vol. 51, n°9, p. 4526-4542.
- Costa, O. L. V., Fragoso, M. D., and Marques, R. P., (2005). Discrete time Markov jump linear systems, New York, Springer-Verlag.
- Doucet, A., Gordon, N. J., and Krishnamurthy, V. (2001). Particle filters for state estimation of jump Markov linear systems, *IEEE Trans. on Signal Processing*, vol. 49, p. 613-624.
- Derrode S. and Pieczynski W. (2013). Exact fast computation of optimal filter in Gaussian switching linear systems, *IEEE Signal Processing Letters*, vol. 20, n°7, p. 701-704.
- Elliot, R. J. and Miao, H. (2006). Stochastic volatility model with filtering, *Stochastic Analysis and Applications*, vol. 24, n°3, p. 661-683.
- Fox, E. Sudderth, E. B. Jordan, M. I. and Willsky, A. S. (2011). Bayesian nonparametric inference of switching dynamic linear models, *IEEE Trans. on Signal Processing*, vol. 59, n°4, p. 1569-1585.
- Giordani P., Kohn, R. and van Dijk, D. (2007). A unified approach to nonlinearity, structural change, and outliers, *Journal of Econometrics*, vol. 137, p. 112-133.
- Johnson L. D. and Sakoulis G. (2008). Maximizing equity market sector predictability in a Bayesian time-varying parameter model, *Computational Statistics & Data Analysis*, vol. 52, n°6, p. 3083-3106.
- Kim C.-J. and Nelson C. R. (1999). State-space models with regime switching, Cambridge, MA: MIT Press.
- Lanchantin P., Lapuyade-Lahorgue J. and Pieczynski W. (2008). Unsupervised segmentation of triplet Markov chains hidden with long-memory noise, *Signal Processing*, vol. 88, n°5, pp. 1134-1151.
- Lanchantin P., Lapuyade-Lahorgue J. and Pieczynski W. (2011). Unsupervised segmentation of randomly switching data hidden with non-Gaussian correlated noise. *Signal Processing*, vol. 91, n°2, p. 163-175.
- Lapuyade-Lahorgue J. and Pieczynski W. (2012). Unsupervised segmentation of hidden semi-Markov non stationary chains, *Signal Processing*, vol. 92, n°1, p. 29-42.
- Le Cam, S. (2013). Identification de distributions dans des series temporelles multivariées présentant des correlation à délais. *actes du 24^{ème} colloque Gretsi*, Brest, France, 3-6 septembre 2013.
- Lee K. Y. and Rheem J. (2000). Smoothing approach using forward-backward Kalman filter with Markov switching parameters for speech enhancement, *Signal Processing*, vol. 80, n°12, p. 2579-2588.
- Petetin, Y. and Desbouvries F. (2013). Un nouvel algorithme de filtrage dans les modèles de Markov à saut linéaires et Gaussiens, *actes du 24^{ème} colloque Gretsi*, Brest, France, 3-6 septembre 2013.
- Petetin, Y. (2013). Algorithmes de restauration Bayésienne mono- et multi- objet(s) dans des modèles Markoviens, thèse de doctorat de Telecom SudParis et de l'Université Pierre et Marie Curie (Paris 6), soutenue le 27 novembre 2013.
- Pieczynski W. and Desbouvries F. (2003). Kalman filtering using Pairwise Gaussian Models, *Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Hong-Kong.
- Pieczynski W. and Abbassi N. (2009). Exact filtering and smoothing in short or long memory stochastic switching systems, *IEEE Int. Workshop on Machine Learning for Signal Processing (MLSP'09)*, September 2-4, Grenoble, France.
- Pieczynski W. (2007). Multisensor triplet Markov chains and theory of evidence, *Int. Journal of Approximate Reasoning*, vol. 45, n°1, p. 1-16.
- Pieczynski W. (2011a). Exact filtering in conditionally Markov switching hidden linear models, *Comptes Rendus Mathématique*, vol. 349, n°9-10, p. 587-590.
- Pieczynski W. (2011b). Exact smoothing in hidden conditionally Markov switching linear models, *Communications in Statistics - Theory and Methods*, vol. 40, n°16, p. 2823 - 2829.
- Pieczynski W. (2013a). Exact calculation of optimal filter in hidden Markov switching long-memory chains, in *Essays on Mathematics and Statistics: Volume 3*, Athens Institute for Education and Research, Vladimir Akis, Editor, 2013.
- Pieczynski, W., Derrode, S., Abbassi, N., Petetin, Y., and Desbouvries, F. (2013b). Exact optimal filtering in an approximating switching system, *Ateliers Traitement et Analyse de l'Information - Méthodes et Applications (TAIMA)*, Hammamet, Tunisie.
- Ramasso, E. and Denoeux, T. (2013). Making use of partial knowledge about hidden states in HMMs: an approach based on belief functions, *IEEE Trans. on Fuzzy Systems*, to appear.
- Ristic, B. Arulampalam, S. and Gordon N. (2004). Beyond the Kalman filter: particle filters for tracking applications, Artech House, Boston, Massachusetts.

- Smith, J. F., Pillai, A., Chen K., and Horwitz B. (2010). Identification and validation of effective connectivity networks in functional magnetic resonance imaging using switching linear dynamic systems, *NeuroImage*, vol. 52, n°3, p. 1027-1040.
- Soo K. N., Kang T. G., Kang S. J., Han C. W., and Hong D. H. (2012). Speech feature mapping based on switching linear dynamic system, *IEEE Trans. On Audio Speech and Language*, vol. 20, n°2, p. 620-631.
- Wu, W. Black, M. J. Mumford, D. Gao, Y. Bienenstock, E. and Donoghue, J. P. (2004). Modeling and decoding motor cortical activity using a switching Kalman filter, *IEEE Trans. on Biomedical Engineering*, vol. 51, n°6, p. 933-942.
- Yin J. J., Zhang J. Q., and Zhao J. (2010). CDF-KF Algorithm for conditionally linear Gaussian state space models, *2nd IEEE Int. Conf. on Advanced Computer Control Location*, Shenyang, China, March 27-29.
- Yu, C. and Zhang, J. (2011). Bayesian approach to Markov switching stochastic volatility model with jumps, *Communications in Statistics – Simulation and Computation*, vol. 40, n°10, p. 1613-1626.
- Zhou T. and Shumway R. (2008). One-step approximations for detecting regime changes in the state space model with application to the influenza data, *Computational Statistics & Data Analysis*, vol. 52, n°5, p. 2277-2291.
- Zoeter O., and Heskes, T. (2006). Deterministic approximate inference techniques for conditionally Gaussian state space models, *Statistical Computation*, vol. 16, p. 279-292.