Filtering in Gaussian linear systems with fuzzy switches
Zied Bouyahia, Stéphane Derrode, Wojciech Pieczynski

To cite this version:
Zied Bouyahia, Stéphane Derrode, Wojciech Pieczynski. Filtering in Gaussian linear systems with fuzzy switches. IEEE Transactions on Fuzzy Systems, Institute of Electrical and Electronics Engineers, 2020, 28 (8), pp.1760-1770. 10.1109/TFUZZ.2019.2921944 . hal-02912919

HAL Id: hal-02912919
https://hal.archives-ouvertes.fr/hal-02912919
Submitted on 7 Aug 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Filtering in Gaussian Linear Systems with Fuzzy Switches

Zied Bouyahia, Stéphane Derrode, and Wojciech Pieczynski

Abstract—This work extends recent results on Conditionally Gaussian Observed Markov Switching Models (CGOMSMs) by incorporating fuzzy switches in the model, instead of hard ones. This kind of generalization is of interest for applications involving continuous switching regimes, such as tracking an object using cameras in intermittent sunlight and shadow conditions. The filter developed hereby is recursive, optimal and exact, up to an approximation of integrals according to some fuzzy measure. Experiences on simulated and on real data – dealing with outdoor air temperature and power consumption of a building – confirm the accuracy and effectiveness of the proposed filter compared to the hard filter with ‘crisp’ switches.

Index Terms—Triplet Markov models, Fuzzy switching linear model, Fast filtering.

NOMENCLATURE

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMSHLM</td>
<td>Conditionally Markov switching hidden linear model.</td>
</tr>
<tr>
<td>CGMSM</td>
<td>Conditionally Gaussian Markov switching Model.</td>
</tr>
<tr>
<td>CGOMSM</td>
<td>Conditionally Gaussian observed Markov switching model, a CGMSM where ( \mathbb{P} ) holds.</td>
</tr>
<tr>
<td>CGOFMSM</td>
<td>Conditionally Gaussian observed fuzzy Markov switching model, a CGMSM with fuzzy switches.</td>
</tr>
<tr>
<td>( X^N_n )</td>
<td>A stochastic process of size ( N ).</td>
</tr>
<tr>
<td>( X_n, x_n )</td>
<td>A random variable at time index ( n ), and a realization.</td>
</tr>
<tr>
<td>( X^N, Y^N, R^N )</td>
<td>State, observation and switches (also called jump) processes, respectively. Denotes ( (X_n, Y_n)^T ) and ( (X_n, R_n, Y_n)^T ), respectively.</td>
</tr>
<tr>
<td>( Z_n, T_n )</td>
<td>The presence of switches. Three stochastic sequences are involved: ( X^N, Y^N, R^N ).</td>
</tr>
<tr>
<td>( K )</td>
<td>Number of switches.</td>
</tr>
<tr>
<td>( m, q )</td>
<td>Dimension of the states and observations.</td>
</tr>
<tr>
<td>( \mathbb{E} [X_n</td>
<td>Y^N_1 = y^N_1] )</td>
</tr>
<tr>
<td>( M_n(r_n) )</td>
<td>Denotes ( \mathbb{E} \left[ \left. (X_n, Y_n)^T \right</td>
</tr>
<tr>
<td>( \nu = \nu_0 + \delta_1 + \mu_{[0,1]} )</td>
<td>Fuzzy measure on ([0,1]) used, where ( \nu_0 ) is the Dirac mass, and ( \mu ) the Lebesgue measure.</td>
</tr>
<tr>
<td>( \nu \otimes \nu )</td>
<td>Denotes the product of measures.</td>
</tr>
</tbody>
</table>

I. INTRODUCTION

Let us consider the problem of statistical optimal filtering in the presence of switches. Three stochastic sequences are involved: states \( X^N = (X_1, \ldots, X_N) \), switches \( R^N = (R_1, \ldots, R_N) \), and observations \( Y^N = (Y_1, \ldots, Y_N) \). For each \( n = 1, \ldots, N \), the random variables \( X_n \) and \( Y_n \) take their values in \( \mathbb{R}^m \) and \( \mathbb{R}^q \) respectively, while \( R_n \) takes its values in the finite discrete set \( \Omega = \{0, 1, \ldots, K - 1\} \). For the sake of simplification, we will assume in the remainder of this paper that (i) \( m = q = 1 \), i.e. \( X^N \) and \( Y^N \) are scalar-valued processes, and (ii) that \( K = 2 \). We consider these hypotheses only to simplify the presentation of the algorithm; the algorithms proposed in the following can be extended to the cases of vectorial processes and for a number of switches greater than two.

The problem is to sequentially estimate each \( X_{n+1} \) from \( Y^N_{n+1} \). Fast recursive optimal filters compute the estimated \( \hat{x}_{n+1}(y^N_{n+1}) = \mathbb{E} \left[ X_{n+1} | Y^N_{n+1} = y^N_{n+1} \right] \) from \( \hat{x}_n(y^N) \) and \( y_{n+1} \). The “Conditionally Gaussian Linear State-Space Models” (CGLSSM), considered as a natural way to extend Gaussian systems to Gaussian switching ones, do not allow for a filtering scheme that can be performed in a reasonable running time [1]–[5]. Classically, CGLSSMs rely on the following assumptions:

1. \( R^N \) is Markov;
2. \( X^N \) is Markov conditionally on \( R^N \);
3. \( (Y_n), 1 \leq n \leq N \), are Gaussian, independent conditionally on \( (R^N, Y^N) \) and verify:
   \[
   p \left( y_n | y^N, x^N \right) = p \left( y_n | r_n, x_n \right) .
   \]

In CGLSSMs, \( R^N \) and \( (R^N, Y^N) \) are both Markov, and \( p \left( y_n | r^N_n, x^N_n \right) \) is very simple. These assumptions do not allow for exact computation of recursive filters, since \( (R^N, Y^N) \) is not Markov and \( p \left( r_n | y^N_n \right) \) cannot be computed sequentially and exactly. This problem has been addressed in recent “conditionally Markov switching hidden linear models” (CMSHLMs [6]), in which both \( R^N \) and \( (R^N, Y^N) \) are Markov, and \( p \left( x_{n+1} | r^N_n, x^N_n \right) \) is pretty general. Here we consider particular Gaussian CMSHLMs called “Conditionally Gaussian Observed Markov Switching Models” (CGOMSMs [7]–[9]), which verify:

1. \( R^N \), \( (R^N, Y^N) \) and \( (X^N, R^N, Y^N) \) are Markov;
2. \( (X^N, Y^N) \) is Gaussian conditionally on \( R^N \).

Figure 1 illustrates the dependencies between the stochastic processes defining the studied CGOMSM. Such an approach is different from classic ones since the hidden chain \( (R^N, X^N) \) is no longer assumed.
Markov, as it has been usually done. Thus, recursive exact filtering is feasible in CGOMSMs and the interesting point is that these models can be quite close to the classic CGLSSMs \[7, 10\].

The values $R_{n+1} = r_{n+1}^{n+1}$ govern the parameters of the distribution

$$p\left(z_{n+1}, y_{n+1} | x_n, r_{n+1}^{n+1}, y_n\right)$$

and thus there exist four possible transitions according to $r_{n+1}^{n+1} \in \Omega^2 = \{0, 1\}^2$. The main limitation of the classical switching model which relies on hard switches is that it does not take into account the transient transition between switches. In real world applications (c.f. Section III), the crisp transition can cause the model to discard significant information that corresponds to the time period during which the system switches from one regime to another. Therefore, the hard switches modeling impacts the accuracy of the filtering scheme.

If we consider the example of tracking a moving object using sensors in intermittent sunlight and shadow conditions (both corresponding to hard switches 0 and 1), there exist intermediary situations as sunlight condition may fade into shadow in a continuous manner. Considering only the hard switches imposes the filtering method to consider only one of the states 0 and 1 during the transition and consequently this shall compromise the accuracy of the filtering since the system state during the transitory phase is a mixture of the two hard switches. Thus, a more complete model would be to associate a set of parameters to each $r_{n+1} \in \Omega = \{0, 1\}$. This is the very aim of the paper: we extend the Markov chain $R_N$ used in CGOMSMs to a ‘fuzzy’ one. Such models called ‘fuzzy models’, have been proposed in \[12\] in a simple context, without Markovianity, to deal with fuzzy image segmentation. Then, they have been extensively used in hidden Markov chains \[13, 14\] and hidden Markov fields \[13, 15, 16\]. Here, the hidden fuzzy Markov chain will be considered to model the pair (switches, observations) in the context of filtering in presence of jumps, which is possible as the pair $(R_N, Y_N)$ is Markov in CGOMSMs.

To the best of our knowledge, while there exist several research works dealing with fuzzy Markov models, the literature for fuzzy Markov jump model filtering is surprisingly scant despite its practical potential. Recently, the discrete-time Takagi-Sugano (T-S.) approach to fuzzy filter design for Markovian jump model has been gaining a significant interest over the last few years –see for example \[17\], \[18\] and the references therein– especially in the fuzzy control and fault detection research community. Unlike T-S. model, our approach assumes that, conditionally to jumps, the model is pairwise linear, which enriches the classical linear models. The proposed fuzzy filter allows for exact calculations for the filter, up to numerical approximation of some integrals, as precise in the text. In the remaining, we start with a brief description of the CGOMSM in Section II and pursue with the description of the original fuzzy jump model in Section III. Sections IV and V depict how the corresponding ‘fuzzy’ filter runs. Section VI reports experimental results that show how the fuzzy filter can improve filtering from the CGOMSM, while Section VII reports comparative filtering result on real data.

II. CONDITIONALLY GAUSSIAN OBSERVED MARKOV HARD SWITCHING MODEL

Let us set $Z_n = (X_n, Y_n)^T$, $T_n = (X_n, R_n, Y_n)^T$ and assume the following:

1) $T_1 = \mathbb{E}$ is Markov;
2) $p(r_{n+1} | t_n) = p(r_{n+1} | r_n)$, which implies the Markovianity of $R_N$;
3) $Z_1 = (Z_1, \ldots, Z_N)$ is Gaussian conditionally on $R_N$.

Such a model, introduced in \[8\], is called “Conditionally Gaussian Markov Switching Model” (CGMSM), and is defined by $p(t_1)$, transitions $p(r_{n+1} | r_n)$, and

$$Z_{n+1} = A_{n+1}(r_{n+1})Z_n + B_{n+1}(r_{n+1})W_{n+1} + N_{n+1}(r_{n+1}),$$

for $n = 1, \ldots, N - 1$, and where

- $W_n = (W_n, V_n)^T$ with $U_1, V_1, \ldots, U_N, V_N$ Gaussian zero-mean independent vectors with identity covariance matrices;
- Matrices $A_{n+1}(r_{n+1})$ and $B_{n+1}(r_{n+1})$:

$$A_{n+1}(r_{n+1}) = \begin{bmatrix} a_{n+1}^{r_{n+1}} & a_{n+1}^{r_{n+1}} \\ a_{n+1}^{r_{n+1}} & a_{n+1}^{r_{n+1}} \end{bmatrix},$$

$$B_{n+1}(r_{n+1}) = \begin{bmatrix} b_{n+1}^{r_{n+1}} & b_{n+1}^{r_{n+1}} \\ b_{n+1}^{r_{n+1}} & b_{n+1}^{r_{n+1}} \end{bmatrix};$$

- Means $N_{n+1}(r_{n+1}) = (N_{n+1}^{X}(r_{n+1}), N_{n+1}^{Y}(r_{n+1}))^T$ are given by

$$N_{n+1}(r_{n+1}) = M_{n+1}(r_{n+1}) - A_{n+1}(r_{n+1})M_n(r_n),$$

with

$$M_n(r_n) = \mathbb{E}\left[ (X_n, Y_n) | r_n \right] = \begin{bmatrix} M_n^X(r_n) \\ M_n^Y(r_n) \end{bmatrix}.$$

Recurrent filtering is not workable in the general CGMSM. Besides, let us notice that the classic “Conditionally Gaussian linear state-space model” (CGLSSM \[2\]) is a particular CGMSM obtained by setting, for each $r_{n+1} \in \Omega^2$

$$a_{n+1}^{r_{n+1}} = a_{n+1}^{r_{n+1}} = a_{n+1}^{r_{n+1}} = a_{n+1}^{r_{n+1}} = 0. \tag{4}$$

Moreover, another particular CGMSM, called “Conditionally Gaussian observed Markov switching model” (CGOMSM) obtained from CGMSM by setting, for each $r_{n+1} \in \Omega^2 \setminus \{0, 1\}$

$$a_{n+1}^{r_{n+1}} = a_{n+1}^{r_{n+1}} = a_{n+1}^{r_{n+1}} = a_{n+1}^{r_{n+1}} = 0, \tag{5}$$

allows for recursive optimal filtering even with switches \[8\]. Indeed, CGOMSM belongs to the category of “conditionally Markov switching linear models” (CMSHLMs) in which recursive optimal filtering is workable \[6\].

The aim of this paper is to extend the CGOMSM defined by \[1\]-\[4\] and \[5\] to a ‘fuzzy’ CGOMSM (denoted by CGOFMSM) and to show how the related recursive optimal ‘fuzzy’ filter runs.

III. CONDITIONALLY GAUSSIAN OBSERVED FUZZY MARKOV SWITCHING MODEL

Let us begin by illustrating with three examples the interest of the new proposed model in real situations.

In the first example, let sequence $X_N$ model the positions at time index $1, \ldots, N$ of a flying object, and let sequence $Y_N$ model the measurements provided by some optical sensors situated on the ground. During the tracking process, the sunlight can be partially or totally hidden due to the presence of clouds, which gives two models for the distribution of $Y_N$. This can be modelled by a ‘hard’ model with $R_N$ such that each $R_n$ takes its value in $\Omega = \{0, 1\}$, with 0 corresponds to total sunlight exposure and 1 to shadow condition. In some situations, during cloudy weather conditions that hide the sun partially, the transition from sunlight to shadow is ‘continuous’, and the duration of ‘intermediary’ light can be of paramount importance to the tracking process. This motivates the introduction of ‘fuzzy’ model with each $R_n$ belonging to $\Omega = \{0, 1\}$ rather than to $\Omega = \{0, 1\}$. However, the distribution of $R_n$ on $\Omega = \{0, 1\}$ has to verify some properties. Willing to have non-null probability to have sunshine - and likewise for shadow - implies that there should be two Dirac masses on 0 and 1. Then one can completely describe the distribution of $R_n$.
on $\Omega = [0, 1]$ by setting continuous probability on $[0, 1]$. Finally, the distribution of $R_n$ is defined by its density $p : [0, 1] \rightarrow \mathbb{R}$ with respect to $\nu = \delta_0 + \delta_1 + \mu_{[0,1]}$, where $\delta_0, \delta_1$ are Dirac’s distributions on 0, 1, and $\mu_{[0,1]}$ is the Lebesgue’s measure on $[0, 1]$.

Let us consider a second example dealing with pedestrian tracking for surveillance purposes which consists in tracking the movements of pedestrians using aggregated data acquired from deployed sensors in the monitored area [19]. Due to the dynamic aspect of pedestrian motion in the presence of several contextual information such as crowd, the use of a two-motion model (corresponding to crowded / uncrowded configurations) is necessary. However, the concept of ‘crowd’ can be seen as a fuzzy phenomenon. Hence, relying on "crowd, the use of a two-motion model (corresponding to crowded configurations) is necessary. However, the concept of ‘crowd’ can be seen as a fuzzy phenomenon. Hence, relying on..."
Recalling that \( R_{n+1} \) and \( Z_n \) are independent conditionally on \( R_n \) (Condition 2 in the definition of CGFSSM), we have
\[
E \left[ Z_{n+1} r_n^{n+1}, y_1^n \right] = \left[ E \left[ X_n \right] r_n, y_1^n \right].
\]

Also, using \( \otimes \) and from classical calculations detailed in \([3]\) we have
\[
\text{Var} \left[ Z_{n+1} r_n^{n+1}, y_1^n \right] = B_{n+1} (r_n^{n+1}) B_{n+1} (r_n^{n+1}) + A_{n+1} (r_n^{n+1}) \text{Var} \left[ Z_n r_n^{n+1}, y_1^n \right] A_{n+1} (r_n^{n+1})
\]
\[
= B_{n+1} (r_n^{n+1}) B_{n+1} (r_n^{n+1}) + A_{n+1} (r_n^{n+1}) \text{Var} \left[ Z_n r_n^{n+1}, y_1^n \right] A_{n+1} (r_n^{n+1}).
\]

For the later convenience, let us note:
\[
\text{Var} \left[ Z_{n+1} r_n^{n+1}, y_1^n \right] = \left[ \alpha_{n+1} + (r_n^{n+1}) \beta_{n+1} (r_n^{n+1}) \delta_{n+1} (r_n^{n+1}) \right].
\]

3) From the multivariate normal distribution specified by \([15]\) and \([16]\), compute the parameters \( E \left[ X_{n+1} r_n^{n+1}, y_1^n \right] \) and \( \text{Var} \left[ X_{n+1} r_n^{n+1}, y_1^n \right] \) of its marginal \( X_{n+1} r_n^{n+1} \) (see \([21]\) Section 8.1.3, page 40) according to
\[
E \left[ X_{n+1} r_n^{n+1}, y_1^n \right] = E \left[ X_{n+1} r_n^{n+1}, y_1^n \right] + \delta_{n+1} (r_n^{n+1}) \xi_{n+1} (r_n^{n+1});
\]
\[
(\xi_{n+1} - E \left[ X_{n+1} r_n^{n+1}, y_1^n \right]) \cup \text{Var} \left[ X_{n+1} r_n^{n+1}, y_1^n \right],
\]

with \( E \left[ X_{n+1} r_n^{n+1}, y_1^n \right] \) and \( \text{Var} \left[ X_{n+1} r_n^{n+1}, y_1^n \right] \) given by eq. \([15]\) and \([16]\).

4) Compute \( E \left[ X_{n+1} r_n^{n+1}, y_1^n \right] \) and \( \text{Var} \left[ X_{n+1} r_n^{n+1}, y_1^n \right] \) using \([14]\) with
\[
E \left[ X_{n+1} r_n^{n+1}, y_1^n \right] = \int_0^1 E \left[ X_{n+1} r_n^{n+1}, y_1^n \right] \ p \left( r_n r_n^{n+1}, y_1^n \right) \ d \nu(r_n)
\]
\[
E \left[ X_{n+1} r_n^{n+1}, y_1^n \right] = \int_0^1 E \left[ X_{n+1} r_n^{n+1}, y_1^n \right] \ p \left( r_n r_n^{n+1}, y_1^n \right) \ d \nu(r_n)
\]

5) Finally, compute the filtering equations
\[
E \left[ X_{n+1} r_n^{n+1}, y_1^n \right] = \int_0^1 E \left[ X_{n+1} r_n^{n+1}, y_1^n \right] \ p \left( r_n r_n^{n+1}, y_1^n \right) \ d \nu(r_n)
\]
\[
E \left[ X_{n+1} r_n^{n+1}, y_1^n \right] = \int_0^1 E \left[ X_{n+1} r_n^{n+1}, y_1^n \right] \ p \left( r_n r_n^{n+1}, y_1^n \right) \ d \nu(r_n)
\]

Remark 1. Integration with respect to \( \nu \) above cannot be written in a closed-form formula. It is, then, approximated by numerical integration. Let \( F \) denote the number of discrete steps used to compute integrals on \([0,1]\). The impact of \( F \) on the restoration results will be discussed in the experimental section.

V. MODEL PARAMETRIZATION

In order to assess the interest of the filtering algorithm on CGOFSSM simulated data, we will consider stationary CGOFSSM models, with distribution defined by \( p \left( x_1^2, z_1^2, y_1^2 \right) = \frac{p \left( z_1^2 \right)}{p \left( x_1^2 \right)} \). Thus, we have to define \( p \left( z_1^2 \right) \) (Section V-A) and \( p \left( x_1^2, y_1^2, z_1^2 \right) \) (Section V-B). Thanks to the particular structure of \( \nu \), this can be done in such a way that when the fuzziness disappears, it is to say when \( p \left( r_n \right) = 0 \) on \([0,1]\) for \( n = 1, \ldots, N \), a CGOFSSM becomes a classical CGFSSM.

A. Distribution of \( (R_1, R_2) \)

Let us notice that in the ‘hard’ case with two possible switches, the distribution \( P_{(R_1, R_2)} \) is simply a probability over \([0,1]^2\). In the fuzzy case we deal with, it is a distribution on \([0,1]^2\), which provides a wide range of possibilities for choosing its shape. We next describe two possible shapes of interest for \( P_{(R_1, R_2)} \) (called FMC1 and FMC2 models, where FMC stands for ‘Fuzzy Markov Chain’), that will be experimented in next Section.

1) First case (FMC1 model): The density \( p \left( r_1^2 \right) \) of \( P_{(R_2)} \ w.r.t. \nu \otimes \nu - \nu \) - where \( \nu = \delta_0 + \delta_1 + \mu_{0,1} - \) is of the form:
\[
p(0,0) = p(1,1), \]
\[
p(0,0) = p(0,1), \]

with
\[
p(r_1, r_2) = \eta + (\delta - \eta) |r_1 - r_2|,
\]

for \( r_1, r_2 \in [0,1]^2 \). A possible shape for this density is illustrated in Fig. 2.

Remark 2. We obtain a ‘fuzzy constant’ model by setting \( \delta = \eta \) and, in particular, we get a ‘purely hard’ CGFSSM model by setting \( \delta = 0 \).

The density \( p \left( r_1 \right) \) of \( P_{(R_1)} \) is computed as follows:
\[
p(r_1) = \begin{cases} \alpha + \beta + \frac{3+2}{2}, & \text{if } r_1 = 0, \end{cases}
\]
\[
p(r_1) = \begin{cases} \alpha + \beta + \frac{3+2}{2}, & \text{if } r_1 = 1, \end{cases}
\]

Knowing that \( \int_0^1 p(r_1) \ d \nu(r_1) = 1 \), we get
\[
\beta = 1 - \frac{1 - 2}{2} (\delta + \eta) + \frac{1}{2} (\delta - \eta) - \alpha.
\]

Hence, this model is only parametrized by \( \{\alpha, \delta, \eta\} \), the calculations are detailed in Appendix [A].

The limit proportion of hard data \( p_H \) with respect to fuzzy ones \( p_F \) in a sampled sequence is
\[
p_H = p(0) + p(1) = 2(\alpha + \beta + \delta, \eta),
\]
\[
p_F = 1 - p_H = \frac{1}{2} (\delta - \eta) - \frac{1}{2} (\delta - \eta).
\]

The density \( p \left( r_2 \right) \) of \( P_{(R_2)} \) is computed as follows:
\[
p(r_2) = \begin{cases} \frac{1}{\eta}, & \text{if } r_2 = 0, \end{cases}
\]
\[
p(r_2) = \begin{cases} \frac{1}{\eta}, & \text{if } r_2 = 1, \end{cases}
\]

if \( r_1 \) and \( r_2 \) are independent, we get
\[
p(r_2) = \frac{\eta + (\delta - \eta) r_2}{\eta + (\delta - \eta) r_2}.
\]

if \( r_2 \in [0,1] \).
FILTERING IN GAUSSIAN LINEAR SYSTEMS WITH FUZZY SWITCHES

5

transient fuzzy changes.  

\int \delta

is illustrated in Fig. 2. By varying

p \in \mathcal{D} \in \mathbb{R}_+^2 \setminus \{0, 1\}^2 \text{ and } -\delta \leq r_2 - r_1 \leq \delta,

and under the constraint that

\int p(r_1, r_2) \, dr_1 \, dr_2 = 1. A possible shape for this density is illustrated in Fig. 2. By varying \delta, this model allows expressing transient fuzzy changes.

Remark 3. If \alpha + \beta = \frac{1}{2}, then \gamma = 0, and the joint law is only made of the four Dirac’s distributions at the four corners, which gives a CGOMSM.

The density \( p(r_1) \) of \( P_{R_1} \) is computed as follows:

\begin{align*}
p(r_1) &= \begin{cases} 
\alpha + \beta + \gamma \delta & \text{if } r_1 = 0, \\
\gamma (\delta + r_1 + 1) & \text{if } r_1 \in [0, \delta], \\
2 \gamma \delta & \text{if } r_1 \in [\delta, 1 - \delta], \\
\alpha + \gamma \delta & \text{if } r_1 = 1.
\end{cases}
\end{align*}

(31)

Since \( \int_{[0,1]} p(r_1) \, dr_1 = 1 \), we get

\begin{align*}
\beta &= 1 - \gamma M - \alpha, 
\end{align*}

(32)

with \( M = \delta (6 - \delta) \), and under the constraint that \( \gamma \leq \frac{1 - 2 \alpha}{\delta} \). Hence this model is only parametrized by \((\alpha, \gamma, \delta)\).

The limit proportion of hard data (\( p_{H} \)) to fuzzy ones (\( p_{F} \)) in a sampled sequence is

\begin{align*}
p_{H} &= p(0) + p(1) = 2(\alpha + \beta + \gamma \delta), \\
p_{F} &= 1 - p_{H} = \gamma \delta (4 - \delta).
\end{align*}

(33)

The density \( p(r_2|r_1) \) of distribution \( P_{R_2|R_1} \), w.r.t. \( \nu \), is the ratio between the joint density and the marginal density. Similarly to the first case, we have to distinguish between different configurations, according to the value of \( r_1 \):

\begin{align*}
p(r_2|r_1 = 0) &= \begin{cases} 
\frac{\alpha}{\alpha + \beta + \gamma \delta} & \text{if } r_2 = 0, \\
\frac{\alpha + \gamma \delta}{\alpha + \beta + \gamma \delta} & \text{if } r_2 = 1.
\end{cases} 
\end{align*}

(34)

\begin{align*}
p(r_2|r_1 \in [0, \delta]) &= \begin{cases} 
\frac{1}{\alpha + \beta + \gamma \delta} & \text{if } r_2 = 0, \\
\frac{1}{\alpha + \gamma \delta} & \text{if } r_2 \in [0, r_1 + \delta], \\
0 & \text{elsewhere.}
\end{cases}
\end{align*}

(35)

\begin{align*}
p(r_2|r_1 \in [\delta, 1 - \delta]) &= \begin{cases} 
\frac{\beta}{\alpha + \beta + \gamma \delta} & \text{if } r_2 \in [r_1 - \delta, r_1 + \delta], \\
0 & \text{elsewhere.}
\end{cases}
\end{align*}

(36)

\begin{align*}
p(r_2|r_1 \in [1 - \delta, 1]) &= \begin{cases} 
\frac{\alpha + \gamma \delta}{\alpha + \beta + \gamma \delta} & \text{if } r_2 = 0, \\
\frac{\alpha + \beta + \gamma \delta}{\alpha + \gamma \delta} & \text{if } r_2 \in [1 - \delta, 1], \\
1 & \text{elsewhere.}
\end{cases}
\end{align*}

(37)

\begin{align*}
p(r_2|r_1 = 1) &= \begin{cases} 
\frac{\delta}{\alpha + \beta + \gamma \delta} & \text{if } r_2 = 0, \\
\frac{\beta + \gamma \delta}{\alpha + \beta + \gamma \delta} & \text{if } r_2 \in [1 - \delta, 1], \\
0 & \text{elsewhere.}
\end{cases}
\end{align*}

(38)

Remark 4. In the two examples of fuzzy Markov models, we assume that the distribution \( P_{R_1,R_2} \) is defined by four masses on the corners, i.e. at locations \( \{0, 1\} \times \{0, 1\} \), and a density on the remaining \( [0, 1] \setminus \{0, 1\} \). It is to say that we assume the distributions on the sides of the square \([0, 1] \times [0, 1] \), i.e. the distributions \( P_{R_1,R_2} \), \( P_{R_1,R_0} \), \( P_{R_0,R} \) and \( P_{R_0,R_1} \) for \( R \) in \([0, 1] \), to be identical to the inner density of the square. This is a particular case, used here simply to simplify the parametrization, since we can set them independently.
B. Distributions of \((X^1_l, Y^1_l)\) conditional on \(R^1_l\)

To finalize the description of stationary \((X^1_l, R^1_l, Y^1_l)\), we need to define the 4D Gaussian distributions \(p(z^1_l, y^1_l | r^1_l) = p(z^1_l | r^1_l)\) for \(r_1, r_2 \in [0, 1]\). The means and covariance matrices of the four 

\[ \begin{bmatrix} \mu_{i,j} = E \begin{bmatrix} X_l \ y_l \end{bmatrix} \mid r_1 = i, r_2 = j \end{bmatrix} = \begin{bmatrix} \mu_i \ 
\Gamma_{i,j} = \begin{pmatrix} \sigma_i \, \sigma_j 
\end{pmatrix} \begin{pmatrix} b_i & a_{i,j} & d_{i,j} 
& c_{i,j} & b_j \end{pmatrix} \end{pmatrix} \end{equation} 

are given by

and

with \(d_{i,j} = b_i c_{i,j}\) in order to verify 5, i.e. for the model to be a CGOMSM.

The mean of a ‘fuzzy Gaussians’ with \(r_1 \in [0, 1]\) is defined by linear interpolation of \(M_i\) and \(M_j\):

\[ M_{r_1} = (1 - r_1) M_0 + r_1 M_1, \]

and its covariance matrix with \(r_1, r_2 \in [0, 1]\) is defined by bi-linear interpolation of the ‘hard covariance matrices’ \(\Gamma_{i,j}:

\[ \Gamma_{r_1} = (1-r_1)(1-r_2) \Gamma_{0,0} + r_1 r_2 \Gamma_{1,1} + r_1(1-r_2) \Gamma_{1,0} + r_2(1-r_1) \Gamma_{0,1}. \]

Then, according to 2, we have

\[ Z_{n+1} = A(r_{n+1}^{(1)}) Z_n + B(r_{n+1}^{(1)}) W_{n+1} + N(r_{n+1}^{(1)}), \]

with \(N(r_{n+1}^{(1)}) = M_{r_{n+1}} - A(r_{n+1}^{(1)}) M_{r_n}\) and

\[ A(r_{n+1}^{(1)}) = \begin{bmatrix} a_{r_{n+1}} & c_{r_{n+1}} 
& b_{r_{n+1}} & d_{r_{n+1}} \end{bmatrix} \begin{pmatrix} (\sigma_i)^2 & b_{r_n} 
& (\sigma_j)^2 & d_{r_n} \end{pmatrix}^{-1}, \]

\[ B(r_{n+1}^{(1)}) = \begin{bmatrix} (\sigma_i)^2 
& b_{r_{n+1}} \end{bmatrix} \begin{pmatrix} (\sigma_i)^2 & b_{r_n} 
& (\sigma_j)^2 & d_{r_n} \end{pmatrix}^{-1} \]

\[ -A(r_{n+1}^{(1)}) = \begin{bmatrix} a_{r_{n+1}} 
& c_{r_{n+1}} \end{bmatrix} \begin{pmatrix} b_{r_{n+1}} & d_{r_{n+1}} 
& c_{r_{n+1}} & b_{r_n} \end{pmatrix}^{-1}. \]

Also, for the later use, note that, according to 33, we have

\[ p(z_{n+1} | z_n, r_{n+1}) = N(A(r_{n+1}^{(1)}) (z_n - M_{r_n}) + M_{r_{n+1}} + B(r_{n+1}^{(1)}) B^t(r_{n+1}^{(1)})). \]

Remark 5. Fuzzy means \[41\] and variances \[42\] can be seen as linear interpolations of hard ones. Other kind of interpolations can be used, once they are compatible with the “hard” CGOMSM.

VI. EXPERIMENTAL STUDIES

Experiments below report results of restoration on simulated data -using the two joint laws presented in the previous Section-, in order to measure the quality of the OFSRF restoration algorithm, and the influence of the number of discretization steps \(F\) in approximation of integrals. Comparison is also performed with the ‘hard’ CGOMSM to measure the error in using this model when there exist transient changes in data.
B. Restoration results

The restoration of simulated data is performed according to the OFSRF algorithm detailed in Section IV, from simulated observations only. Fig. 4 shows an example of the restoration of $r_N^1$ and $x_N^1$ for the FMC2 law and when the number of discrete fuzzy jumps are set to $F = 5$. The restoration of $r_N^1$ was obtained by applying Maximum Posteriori Mode (MPM) principle:

- If $p(r_n = 0 | y_n^1) + p(r_n = 1 | y_n^1) > 0.5$ then the restoration will be “hard”, else it will be “fuzzy”.
- If the restoration is “hard”, set $\hat{r}_n$ to be “0” if $p(r_n = 0 | y_n^1) > p(r_n = 1 | y_n^1)$, else set $\hat{r}_n$ to be “1”;
- If the restoration is “fuzzy”, set $\hat{r}_n$ to the discrete fuzzy jump which maximizes $p(r_n | y_n^1)$.

Note however that the restoration of $r_N^1$ is only performed for illustration purpose and is not required for the restoration of $x_N^1$.

We can observe the numerical effect of $F$ with the presence of stair-steps in the restoration of jumps in (b). Fig. (a) shows the restoration of $x_N^1$ assuming that the jumps are known (i.e. using the simulated fuzzy Markov chain for $r_N^1$). Fig. (c) assumes that the jumps are unknown (OFSRF algorithm). Also, fig. (c) allows to observe the restoration difference between the ‘classic’ CGOFMSM and the ‘fuzzy’ CGOFMSM, the latter follows the simulated states better than the first one for $n \leq 30$. This behaviour must be compared with that of the fuzzy jumps in Fig. (b) which shows a large difference in the restoration of jumps when $n \leq 30$.

To measure the quality of restorations with respect to $F$, we compute the mean MSE of 50 independent experiments of $N = 300$ samples, for the FMC1 Markov law. Fig. 5 shows the MSE evolution for increasing values of $F$ for both $x_N^1$ and $r_N^1$. The result of applying the CGOMSM filter on the data is reported in the same graph (horizontal black dotted line, denoted as ‘Hard filter - UJ’). The parameters used for the hard filters are the same than the ones used for the fuzzy filter, except that we ‘harden’ the Markov laws by integrating the fuzzy laws on all 4 quadrants of $[0, 1]$ to get
P(\(R_1, R_2\)). In this figure, we can observe that the excess error over the model with known jumps is halved. Also, the state MSE reaches its minimum when \(F > 3\); this value depends on the fuzzy Markov model and on its parameters. According to some other experiments not reported here, \(F\) is always kept relatively small (typically \(F \leq 5\)), which is of interest since the larger \(F\) is, the more the computing time increases. Indeed, the complexity is linear \(w.r.t.\) to \(F\), which is to say that the computational burden of CGOFMSM is approximately the same as the one of CGOMSM with \(2 + F\) jumps.

The last experiment, whose results are reported in Table I for the FMC1 and FMC2 laws, shows the restoration MSE for the CGOMSM and the CGOFMSM filters when the number of fuzzy samples in the simulated data is decreasing (by adjusting parameter \(\beta\) values). We can observe that, for both fuzzy Markov models, the hard filter reaches the performance of the fuzzy filter only when the percent of hard jumps is near 100%. Elsewhere the fuzzy filter provides lower MSE, and the difference can be very large for hard sample rates lower than 50%. This result confirms the interest of the fuzzy filter in comparison with the hard one in the presence of transient changes in observation data.

### Table I

<table>
<thead>
<tr>
<th>% of ‘hard’ samples - FMC1</th>
<th>48%</th>
<th>58%</th>
<th>67%</th>
<th>75%</th>
<th>93%</th>
</tr>
</thead>
<tbody>
<tr>
<td>jumps MSE (CGOFMSM)</td>
<td>0.051</td>
<td>0.039</td>
<td>0.034</td>
<td>0.017</td>
<td></td>
</tr>
<tr>
<td>states MSE (CGOFMSM)</td>
<td>0.475</td>
<td>0.431</td>
<td>0.415</td>
<td>0.368</td>
<td></td>
</tr>
<tr>
<td>jumps MSE (CGOMSM)</td>
<td>0.126</td>
<td>0.074</td>
<td>0.057</td>
<td>0.020</td>
<td></td>
</tr>
<tr>
<td>states MSE (CGOMSM)</td>
<td>0.599</td>
<td>0.500</td>
<td>0.468</td>
<td>0.386</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>% of ‘hard’ samples - FMC2</th>
<th>48%</th>
<th>62%</th>
<th>75%</th>
<th>87%</th>
<th>100%</th>
</tr>
</thead>
<tbody>
<tr>
<td>jumps MSE (CGOFMSM)</td>
<td>0.030</td>
<td>0.022</td>
<td>0.021</td>
<td>0.015</td>
<td>0.012</td>
</tr>
<tr>
<td>states MSE (CGOFMSM)</td>
<td>0.442</td>
<td>0.420</td>
<td>0.399</td>
<td>0.37</td>
<td></td>
</tr>
<tr>
<td>jumps MSE (CGOMSM)</td>
<td>0.070</td>
<td>0.084</td>
<td>0.073</td>
<td>0.064</td>
<td>0.031</td>
</tr>
<tr>
<td>states MSE (CGOMSM)</td>
<td>0.538</td>
<td>0.457</td>
<td>0.422</td>
<td>0.374</td>
<td></td>
</tr>
</tbody>
</table>

### VIII. Conclusion

We proposed a new jump Markov model made of a triplet random process (observations, hidden states, hidden fuzzy switches), and designed the related optimal recursive fuzzy filter which is able to restore switches and states from observations. We called this model “Conditionally Gaussian Observed Fuzzy Markov switching Model” (CGOFMSM). The work is based on two key ideas:

- A recursive and exact filter to deal with hard jumps, called CGOMSM, is available, see [7]–[10]. This filter only assumes the presence of a zero-term \(s\) in the transition matrix of the very general Conditionally Gaussian Markov Switching model defined by [2].

This time series, collected by the American Department of Energy, is open-data and can be downloaded for free from [https://openei.org/datasets/dataset/consumption-outdoor-air-temperature-11-commercial-buildings](https://openei.org/datasets/dataset/consumption-outdoor-air-temperature-11-commercial-buildings). The experiments focus on building #5.
straightforward, using matrix products. However, the extension of the
filter to three and more classes is not as easy, with quite complex
fuzzy Markov laws to deal with, but could be inspired from the
work [22]. The next step in the development of an unsupervised
parameter estimation method for this CGOFMSM—similar to the one
proposed for the ‘hard’ model [23]—is the derivation of a fuzzy
smoother for off-line processing. Application of the fuzzy model
to deal with the design of a control system for road traffic congestion
prediction in which traffic dynamics would be modeled by a switching
regime model is another perspective for further work.

**APPENDIX**

Here are the detail to specify the margin \( p(r_1) \) and the parameter
\( \beta \) from the joint density \( p(r_1, r_2) \) defined in Section V-A.1

### A. Calculation for margin law \( p(r_1) \)

The density \( p(r_1) \) of distribution \( P_{R_1}, \) w.r.t. \( \nu, \) is obtained by

\[
p(r_1) = \int_0^1 p(r_1, r_2) \, dr_2 = p(r_1, 0) + p(r_1, 1) + \int_0^1 \eta + (\delta - \eta) |r_1 - r_2| \, dr_2,
\]

with

\[
A(r_1) = \eta + (\delta - \eta) \int_0^1 |r_1 - r_2| \, dr_2,
\]

and

\[
B_1(r_1) = \int_0^{r_1} (r_1 - r_2) \, dr_2 = r_1^2 - \frac{1}{2} r_1^2 = \frac{1}{2} r_1^2,
\]

\[
B_2(r_1) = \int_{r_1}^1 (r_2 - r_1) \, dr_2 = \frac{1}{2} (1 - r_1^2) - r_1 (1 - r_1) = \frac{1}{2} (1 - r_1^2) - r_1.
\]

Thus we have \( B(r_1) = \frac{1}{2} r_1^2 - r_1, \) and \( A(r_1) = \eta + (\delta - \eta) (\frac{1}{2} r_1^2 - r_1) \).

So, for \( r_1 = 0, \) \( p(r_1) = \alpha + \beta + A(0) = \alpha + \beta + \frac{\delta + \eta}{2}, \) for \( r_1 = 1, \)
\( p(r_1) = \alpha + \beta + A(1) = \alpha + \beta + \frac{1}{2}, \) and for \( r_1 \in [0, 1[, \)
\( p(r_1) = \eta + (\delta - \eta) r_1 + \eta + (\delta - \eta) (1 - r_1) + \eta + (\delta - \eta) \left( \frac{1}{2} r_1^2 - r_1 \right)
\]

\[= 3 \eta + (\delta - \eta) \left( \frac{3}{2} \right) + (\delta - \eta) (r_1^2 - r_1)
\]

\[= \frac{3}{2} (\delta + \eta) + (\delta - \eta) (r_1^2 - r_1).
\]

Hence, we get eq. 23.

### B. Calculation of \( \beta \)

We have

\[
\int_0^1 p(r_1) \, dr_1 = p(0) + p(1) + \int_0^1 p(r_1) \, dr_1 = 2 (\alpha + \beta) + (\delta + \eta) + \int_0^1 \frac{3}{2} (\delta + \eta) + (\delta - \eta) (r_1^2 - r_1) \, dr_1.
\]

In this work, we assume scalar states and scalar observations for
notation convenience; the extension to a vectorial filter is somewhat

- The definition of a mixed measure including two Dirac masses
for hard classes “0” or “1” and a Lebesgue measure to deal with
fuzziness. It should be noted that integral calculations required
some simple and low-time consuming numerical approximation.

We showed through an experimental study that the proposed model
and its filter provide interesting results in terms of data restoration
accuracy. This behaviour is confirmed when the model in confronted
to real data dealing with outdoor air temperature. In that case, the
fuzzy jumps allow a better modelling of the increasing and decreasing
air temperature cycle during one day.

In this work, we assume scalar states and scalar observations for
notation convenience; the extension to a vectorial filter is somewhat

![Fig. 7. Filtering result with estimated jumps (up) and estimated states (down)
for the hard model with \( F = 5. \) The MSE for jumps is 0.07, while the MSE
for states is 6759.](image)

![Fig. 8. Filtering result with estimated jumps (up) and estimated states (down)
for the hard model. The MSE for jumps is 0.13, while the MSE for states is
8701.](image)
with
\[
C_1 = \int_0^1 \frac{3}{2} (\delta + \eta) \, dr = \frac{3}{2} (\delta + \eta),
\]
\[
C_2 = (\delta - \eta) \int_0^1 (r_1^2 - r_1) \, dr = -\frac{1}{2} (\delta - \eta).
\]
Knowing that \( \int_0^1 p(r_1) \, dr_1 = 1 \), we get \( 2(a + \beta) + (\delta + \eta) + \frac{3}{2} (\delta + \eta) - \frac{1}{2} (\delta - \eta) = 1 \), and find result in eq. \( 16 \).

C. Calculation to get eq. \( 16 \)

From
\[
\text{Var} \left[ r_{n+1}^T, y_n^T \right] = E \left[ Z_{n+1} r_{n+1}^T, y_n^T \right] - E \left[ Z_{n+1} r_{n+1}^T \right] E \left[ Z_{n+1}^T r_{n+1}^T \right],
\]
we have
\[
\begin{align*}
&= A_{n+1} (r_{n+1} - E \left[ Z_{n+1} r_{n+1}^T \right]) + N_{n+1} (r_{n+1} - E \left[ Z_{n+1} r_{n+1}^T \right]) + E \left[ Z_{n+1} r_{n+1}^T \right] - E \left[ Z_{n+1}^T r_{n+1}^T \right] + \mathcal{B} (r_{n+1} - E \left[ Z_{n+1} r_{n+1}^T \right]) + E \left[ Z_{n+1} r_{n+1}^T \right] - E \left[ Z_{n+1}^T r_{n+1}^T \right],
\end{align*}
\]
and, using \( 13 \),
\[
\begin{align*}
&= A_{n+1} (r_{n+1} - E \left[ Z_{n+1} r_{n+1}^T \right]) + N_{n+1} (r_{n+1} - E \left[ Z_{n+1} r_{n+1}^T \right]) + \mathcal{B} (r_{n+1} - E \left[ Z_{n+1} r_{n+1}^T \right]) + E \left[ Z_{n+1} r_{n+1}^T \right] - E \left[ Z_{n+1}^T r_{n+1}^T \right]
\end{align*}
\]
Subtracting \( 47 \) and \( 48 \) gives \( 16 \).

REFERENCES