Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

An Algorithmic Estimation Scheme for Hybrid Stochastic Systems

W. P. Malcolm, R. J. Elliott, F Dufour and M. S. Arulampalam

Abstract— **In this article we describe a state estimation algorithm for discrete-time Gauss-Markov models whose parameters are determined at each discrete-time instant by the state of a Markov chain. The scheme we develop is fundamentally distinct from extant methods, such as the so called Interacting Multiple Model algorithm (IMM) in that it is based directly upon the *exact* hybrid filter dynamics.**

The enduring and well known obstacle in estimation of jump Markov systems, is managing the geometrically growing history of candidate hypotheses. Our scheme maintains a fixed number of candidate paths in a history, each identified by an optimal subset of estimated mode probabilities.

We present here a finite dimensional sub-optimal filter for the information state. Corresponding finite dimensional recursions are also given for the mode probability estimate, the state estimate and is associated state error covariance The memory requirements of our filter are fixed in time. A computer simulation is included to demonstrate performance of the Gaussian-mixture algorithm described.

I. INTRODUCTION

The problem we are concerned with in this article is a pure synthesis problem for a particular class stochastic hybrid system.

This synthesis problem concerns computing a state and mode estimation algorithm whose memory requirements remain fixed in time. The main challenge in such a task, is to balance computational complexity against accuracy of estimation. In this article we take new approach, by developing our estimation schemes upon the foundation of the exact filter dynamics, (see [4]). Using this exact filter, we develop a suboptimal scheme by incorporating finite Gaussian mixture representations for the filter probability densities. Our choice of a Gaussian mixture representation is justified by basic results in [9]. By applying a Lemma for reproducing Gaussian densities (see [5]), we compute a closed form recursive estimation algorithm, whose outputs are: a conditional mean estimate of the hidden state, an estimate of the state error covariance associated with this estimate and an estimate of the mode probability. The basic and omnipresent problem of exponential growth in complexity in the exact filter is circumvented by a scheme for the management of hypotheses. This scheme is an extension of an idea due to Viterbi [12].

The pseudo-code algorithm we present here is developed from the results in the [8]. This article uses change of probability measure techniques, which due to limitations of space, we do not detail here, rather, the contribution of this article is to cast the mathematical results in [8] into a pseudo code form amenable to the practitioner and to demonstrate the benefits of this algorithm through a computer simulation.

II. STOCHASTIC DYNAMICS

All processes are defined, initially, on a fixed probability space $(\Omega, \mathcal{F}, P)$.

A. Markov Chain Dynamics

We consider a time-homogeneous discrete-time m-state Markov chain Z. It is convenient to identify the state space of Z with an orthonormal basis indicator functions, which we denote by $\mathcal{L}$,

$$\mathcal{L} = \{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_m\} = \left\{ \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \ldots, \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \right\} \subseteq \mathbb{R}^m. \tag{1}$$

We suppose our Markov chain has sufficient statistics (Π, p_0), where $\Pi = \left[\pi_{(j,i)}\right]_{\substack{1\leq j\leq m \\ 1\leq i\leq m}}$ is the transition matrix of Z, with elements

$$\pi_{(j,i)} \triangleq P(Z_k = \boldsymbol{e}_j \mid Z_{k-1} = \boldsymbol{e}_i), \quad \forall k \in \mathbb{N} \tag{2}$$

and $E\big[Z_0\big] = p_0$.

B. State Process Dynamics

We suppose the indirectly observed state vector $x \in \mathbb{R}^{n\times 1}$, has dynamics

$$x_k = \sum_{j=1}^{m} \big\langle Z_k, \boldsymbol{e}_j \big\rangle A_j x_{k-1} + \sum_{j=1}^{m} \big\langle Z_k, \boldsymbol{e}_j \big\rangle B_j w_k. \tag{3}$$

Here w is a vector-valued Gaussian process with $w \sim N(0, \boldsymbol{I}_n)$. A_j and B_j are $n \times n$ matrices and for each $j \in \{1, 2, \ldots, m\}$, are nonsingular.

C. Observation Process Dynamics

Consider a vector-valued observation process with values in $\mathbb{R}^{d\times 1}$ and dynamics

$$y_k = \sum_{j=1}^{m} \big\langle Z_k, \boldsymbol{e}_j \big\rangle C_j x_k + \sum_{j=1}^{m} \big\langle Z_k, \boldsymbol{e}_j \big\rangle D_j v_k. \tag{4}$$

Here v is a vector-valued Gaussian process with $v \sim N(0, \boldsymbol{I}_d)$. We suppose the matrices $D_j \in \mathbb{R}^{d\times d}$, for each $j \in \{1, 2, \ldots, m\}$, are nonsingular. The systems we shall consider in this article are described by the dynamics (3) and (4). The three stochastic processes Z, x and y are mutually statistically independent. Taken together, these dynamics form a triply stochastic system, with random inputs due to the processes Z, x and y. For example, if at time k $Z_k = \boldsymbol{e}_j$, then the dynamical model with state x and observation y, is

0-7803-9568-9/05/$20.00 ©2005 IEEE

defined by the parameters set $\{A_j, B_j, C_j, D_j\}$.
We define our filtrations as follows:

$$\mathbb{Y}_k = \{\mathcal{Y}_\ell\}_{0\le\ell\le k} \quad \text{where} \quad \mathcal{Y}_k = \sigma\{y_\ell, 0 \le \ell \le k\}, \tag{5}$$
$$\mathbb{G}_k = \{\mathcal{G}_\ell\}_{0\le\ell\le k} \quad \text{where} \quad \mathcal{G}_k = \sigma\{Z_\ell, x_\ell, y_\ell, 0 \le \ell \le k\}. \tag{6}$$

Notation: To denote the inverse of a matrix A, we write $\operatorname{inv}(A)$.

III. Exact Hybrid filter Dynamics

The exact filter for the hybrid stochastic system defined in the previous section, (given below), first appeared in the article [4]. Despite the important contributions of this article being relatively new, the exact filter of Elliott Dufour and Sworder has been largely overlooked by the tracking community. In contrast, it is important to note that common schemes, such as the so-called IMM (Interacting Multiple Model algorithm, see [2]), are not based upon the exact filter dynamics.

Definition 1 *The symbol $\Phi(\cdot)$ will be used to denote the zero mean normal density on $\mathbb{R}^{d\times 1}$*

$$\Phi(\xi) = (2\pi)^{-d/2}\exp\left(-\tfrac{1}{2}\xi'\xi\right). \tag{7}$$

The exact state estimation filter given in [4] is written in unnormalised form, that is, dynamics satisfied by an unnormalised probability density. These dynamics are computed using reference probability techniques, see [3], [6] and [7]. Briefly, we are interesting in computing conditional probabilities for joint events of the form $P(x \in dx, Z_k = \boldsymbol{e}_j \mid \mathcal{Y}_k)$. Omitting the details, we assume the existence unnormalised probability densities corresponding to our events of interest, where, for example,

$$P(x \in dx, Z_k = \boldsymbol{e}_j \mid \mathcal{Y}_k) = \frac{q_k^j(x)dx}{\displaystyle\int_{\mathbb{R}^n} q_k^j(\xi)d\xi}. \tag{8}$$

What we would like to compute, is recursive dynamics whose solutions are the densities $q^j(x) : \mathbb{R}^{n\times 1} \to \mathbb{R}_+$, where $j \in \{1, 2, \ldots, m\}$. To this end, we recall the fundamental contribution of [4] in the next Theorem.

Theorem 1 *The un-normalised probability density $q_k^j(x)$, satisfies the following integral-equation recursion,*

$$q_k^j(x) = \frac{\Phi\big(D_j^{-1}(y_k - C_j x)\big)}{\Phi(y_k)|D_j||B_j|}\sum_{r=1}^{m}\pi_{(j,r)}\times \int_{\mathbb{R}^n}\Psi\big(B_j^{-1}(x - A_j\xi)\big)q_{k-1}^r(\xi)d\xi. \tag{9}$$

Remark III.1 It is immediate that the dynamics at (1) will grow exponentially in complexity as a function of the discrete time index k.

IV. Gaussian Mixture Densities

Theorem 2 *Suppose the un-normalised probability density $q_{k-1}^r(\xi)$, (as it appears under the integral in equation (9)), can be written as a finite weighted Gaussian mixture with $M^q \in \mathbb{N}$ components. That is, for $k \in \{1, 2, \ldots\}$, we suppose*

$$q_{k-1}^r(\xi) = \sum_{s=1}^{M^q} W_{k-1}^{(r,s)}\frac{1}{(2\pi)^{n/2}|\Sigma_{k-1|k-1}^{(r,s)}|^{\frac{1}{2}}}\times \exp\Big\{-\tfrac{1}{2}(\xi - \alpha_{k-1|k-1}^{(r,s)})'\operatorname{inv}\big(\Sigma_{k-1|k-1}^{(r,s)}\big)(\xi - \alpha_{k-1|k-1}^{(r,s)})\Big\}. \tag{10}$$

Here $\Sigma_{k-1|k-1}^{(r,s)} \in \mathbb{R}_+^{n\times n}$, and $\alpha_{k-1|k-1}^{(r,s)} \in \mathbb{R}^{n\times 1}$, are both $\mathcal{Y}_{k-1}$-measurable functions for all pairs $(r, s) \in \{1, 2, \ldots, m\} \times \{1, 2, \ldots, M^q\}$. Using this Gaussian mixture (10), the equation for the optimal un-normalised density process has the form

$$q_k^j(x) \triangleq \frac{1}{(2\pi)^{(d+n)/2}\Phi(y_k)}\sum_{r=1}^{m}\sum_{s=1}^{M^q}K^{j,(r,s)}\times \exp\Big\{-\tfrac{1}{2}\Big(x - \operatorname{inv}\big(\sigma^{j,(r,s)}\big)\delta^{j,(r,s)}\Big)'\times \sigma^{j,(r,s)}\Big(x - \operatorname{inv}\big(\sigma^{j,r,s}\big)\delta^{j,(r,s)}\Big)\Big\}. \tag{11}$$

A proof of Theorem 2 is given in [8].

The foremost value of the density representation given at (11), is that integrations over the space $\mathbb{R}^{n\times 1}$ have been eliminated. However, this representation also grows exponentially in its complexity as a function of the discrete-time index k. Our objective then, is to develop an approximation to this expression, whose memory requirements are finite and invariant to time. The technical details upon which the following section is based, are well beyond the size limitation of this paper, however, comprehensive detail concerning the basis of our pseudo code algorithm in Section V can be found in [8]. The key step in what follows, is to replace the double summation in equation (10), with a single summation over the *optimal* contributors within the Gaussian mixture. Suppose one could identify M^q optimal pairs of indices in the set $\{1, 2, \ldots, m\} \times \{1, 2, \ldots, M\}$ and then use these indices to construct a suboptimal but fixed in memory requirements density. Further, suppose the measure corresponding to this suboptimal approach is written as $P_{M^q}^*$. Then the state estimate would be determined by the expectation

$$\widehat{x}_{k|k} \triangleq E^{P_{M^q}^*}\big[x_k \mid \mathcal{Y}_k\big]. \tag{12}$$

We now detail an algorithm to evaluate the expectation at (12), its associated state error covariance and the estimated mode probability.

V. Pseudo Code Algorithm

In this section we define a three-step pseudo code form of our estimation algorithm. Following the completion of an initialisation step, our three algorithm steps are:

Step 1 Compute Gaussian-mixture Densities,
Step 2 Hypothesis management,
Step 3 Updating.

In details these steps are given below.

Initialisation

- Choose initial Gaussian mixtures statistics:

$$\mathcal{A}_0 \triangleq \begin{bmatrix} \alpha_{0|0}^{1,1} & \alpha_{0|0}^{1,2} & \dots & \alpha_{0|0}^{1,M^q} \\ \alpha_{0|0}^{2,1} & \alpha_{0|0}^{2,2} & \dots & \alpha_{0|0}^{2,M^q} \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_{0|0}^{m,1} & \alpha_{0|0}^{m,2} & \dots & \alpha_{0|0}^{m,M^q} \end{bmatrix} \quad (13)$$

and

$$\mathcal{B}_0 \triangleq \begin{bmatrix} \Sigma_{0|0}^{1,1} & \Sigma_{0|0}^{1,2} & \dots & \Sigma_{0|0}^{1,M^q} \\ \Sigma_{0|0}^{2,1} & \Sigma_{0|0}^{2,2} & \dots & \Sigma_{0|0}^{2,M^q} \\ \vdots & \vdots & \vdots & \vdots \\ \Sigma_{0|0}^{m,1} & \Sigma_{0|0}^{m,2} & \dots & \Sigma_{0|0}^{m,M^q} \end{bmatrix}. \quad (14)$$

- For each $j \in \{1, 2, \dots, m\}$, choose, by some means, the initial sets of Gaussian-mixture weights $\{W_0^{(j,1)}, W_0^{(j,2)}, \dots, W_0^{(j,M^q)}\}$.

For $k = 1, 2, \dots N$, repeat:

Step 1, Compute Gaussian Mixture Densities:

- Define an index set, $\Gamma \triangleq \{1, 2, \dots, m\} \times \{1, 2, \dots, M^q\}$.
- For each $j \in \{1, 2, \dots, m\}$, at each time k, compute of the following $m \times M^q$ quantities, each indexed by the pairs $(r, s) \in \Gamma$:

$$\overline{\Sigma}_{k-1|k-1}^{j,(r,s)} \triangleq B_j B_j' + A_j \Sigma_{k-1|k-1}^{(r,s)} A_j' \quad (15)$$

$$\widetilde{u}_{k-1|k-1}^{j,(r,s)} \triangleq A_j \alpha_{k-1|k-1}^{(r,s)} \quad (16)$$

$$\sigma^{j,(r,s)} \triangleq C_r' \operatorname{inv}\big(D_r D_r'\big) C_r + \operatorname{inv}\big(\overline{\Sigma}_{k-1|k-1}^{j,(r,s)}\big) \quad (17)$$

$$\operatorname{inv}\big(\sigma^{j,(r,s)}\big) \triangleq \overline{\Sigma}_{k-1|k-1}^{j,(r,s)} - \overline{\Sigma}_{k-1|k-1}^{j,(r,s)} C_r' \operatorname{inv}\Big(C_r \overline{\Sigma}_{k-1|k-1}^{j,(r,s)} C_r' + D_r D_r'\Big) C_r \overline{\Sigma}_{k-1|k-1}^{j,(r,s)} \quad (18)$$

$$\delta^{j,(r,s)} \triangleq \operatorname{inv}(\overline{\Sigma}_{k-1|k-1}^{j,(r,s)}) \widetilde{u}_{k-1|k-1}^{j,(r,s)} + C_r' \operatorname{inv}(D_r D_r') y_k \quad (19)$$

$$K^{j,(r,s)} \triangleq \frac{\pi_{(j,r)} W_{k-1}^{(r,s)}}{|\overline{\Sigma}_{k-1|k-1}^{j,(r,s)}|^{\frac{1}{2}} |D_j|} \times \exp\Big\{\tfrac{1}{2} (\delta^{j,(r,s)})' \operatorname{inv}(\sigma^{j,(r,s)}) \delta^{j,(r,s)}\Big\} \times \exp\Big\{-\tfrac{1}{2}\Big[y_k' \operatorname{inv}(D_r D_r') y_k + (\widetilde{u}_{k-1|k-1}^{j,(r,s)})' \operatorname{inv}(\overline{\Sigma}_{k-1|k-1}^{j,(r,s)}) \widetilde{u}_{k-1|k-1}^{j,(r,s)}\Big]\Big\} \quad (20)$$

Step 2, Hypothesis Management:

- For each $j \in \{1, 2, \dots, m\}$, compute the quantities

$$\mathcal{C}^j \triangleq \begin{bmatrix} \zeta^{j,(1,1)} & \zeta^{j,(1,2)} & \dots & \zeta^{j,(1,M^q)} \\ \zeta^{j,(2,1)} & \zeta^{j,(2,2)} & \dots & \zeta^{j,(2,M^q)} \\ \vdots & \vdots & \vdots & \vdots \\ \zeta^{j,(m,1)} & \zeta^{j,(m,2)} & \dots & \zeta^{j,(m,M^q)} \end{bmatrix}. \quad (21)$$

Here, for example,

$$\zeta^{j,(r,s)} = K^{j,(r,s)} |\sigma^{j,(r,s)}|^{-\frac{1}{2}}. \quad (22)$$

- Using each matrix $\{\mathcal{C}^1, \mathcal{C}^2, \dots, \mathcal{C}^m\}$, compute the optimal index sets

$$\mathcal{I}^1 = \big\{(r_{1,1}^*, s_{1,1}^*), \dots, (r_{1,M^q}^*, s_{1,M^q}^*)\big\} \quad (23)$$

$$\mathcal{I}^2 = \big\{(r_{2,1}^*, s_{2,1}^*), \dots, (r_{2,M^q}^*, s_{2,M^q}^*)\big\} \quad (24)$$

$$\vdots \quad \vdots$$

$$\mathcal{I}^m = \big\{(r_{m,1}^*, s_{m,1}^*), \dots, (r_{m,M^q}^*, s_{m,M^q}^*)\big\} \quad (25)$$

where, for example, the index set $\mathcal{I}^j$ is computed via the successive maximisations:

$$\zeta^{j,(r_{j,1}^*, s_{j,1}^*)} \triangleq \max_{(r,s) \in \Gamma} \{\mathcal{C}^j\}, \quad (26)$$

$$\zeta^{j,(r_{j,2}^*, s_{j,2}^*)} \triangleq \max_{(r,s) \in \Gamma \backslash (r_{j,1}^*, s_{j,1}^*)} \{\mathcal{C}^j\}, \quad (27)$$

$$\vdots \qquad \vdots$$

$$\zeta^{j,(r_{j,M^q}^*, s_{j,M^q}^*)} \triangleq \max_{(r,s) \in \Gamma \backslash \{(r_{j,1}^*, s_{j,1}^*), \dots\}} \{\mathcal{C}^j\}. \quad (28)$$

Step 3, Updating:

- The two estimates $\widehat{x}_{k|k} = E\big[x_k \mid \mathcal{Y}_k\big]$ and $\Sigma_{k|k} = E\big[(x_k - \widehat{x}_{k|k})(x_k - \widehat{x}_{k|k})' \mid \mathcal{Y}_k\big]$, are computed, respectively, by the formulae:

$$
\begin{aligned}
&\widehat{x}_{k|k} = \\
&\frac{\sum_{j=1}^{m}\sum_{\ell=1}^{M^q} \frac{K^{j,(r^*_{j,\ell},s^*_{j,\ell})}}{|\sigma^{j,r^*_{j,\ell},s^*_{j,\ell}}|^{\frac{1}{2}}} \operatorname{inv}\big(\sigma^{j,r^*_{j,\ell},s^*_{j,\ell}}\big)\delta^{j,(r^*_{j,\ell},s^*_{j,\ell})}}{\sum_{j=1}^{m}\sum_{\ell=1}^{M^q} K^{j,(r^*_{j,\ell},s^*_{j,\ell})}|\sigma^{j,(r^*_{j,\ell},s^*_{j,\ell})}|^{-\frac{1}{2}}} \\
&= \left\{\frac{1}{\sum_{j=1}^{m}\sum_{\ell=1}^{M^q} K^{j,(r^*_{j,\ell},s^*_{j,\ell})}|\sigma^{j,(r^*_{j,\ell},s^*_{j,\ell})}|^{-\frac{1}{2}}}\right\}\times \\
&\sum_{j=1}^{m}\sum_{\ell=1}^{M^q}\Bigg\{\frac{K^{j,(r^*_{j,\ell},s^*_{j,\ell})}}{|\sigma^{j,r^*_{j,\ell},s^*_{j,\ell}}|^{\frac{1}{2}}}\times \\
&\Bigg[A_{r^*_{j,\ell}}\alpha^{r^*_{j,\ell},s^*_{j,\ell}}_{k-1|k-1} + \\
&\overline{\Sigma}^{j,r^*_{j,\ell},s^*_{j,\ell}}_{k-1|k-1} C'_{r^*_{j,\ell}} \operatorname{inv}\Big(D_{r^*_{j,\ell}}D'_{r^*_{j,\ell}} + \\
&C_{r^*_{j,\ell}}\overline{\Sigma}^{j,(r^*_{j,\ell},s^*_{j,\ell})}_{k-1|k-1}C^{-1}_{r^*_{j,\ell}}\Big)\times \\
&\big(y_k - C_{r^*_{j,\ell}}A_{r^*_{j,\ell}}\alpha^{r^*_{j,\ell},s^*_{j,\ell}}_{k-1|k-1}\big)\Bigg]\Bigg\}
\end{aligned}
\tag{29}
$$

and

$$
\begin{aligned}
&\widehat{\Sigma}_{k|k} = \\
&\left\{\frac{1}{\sum_{j=1}^{m}\sum_{\ell=1}^{M^q} K^{j,(r^*_{j,\ell},s^*_{j,\ell})}|\sigma^{j,(r^*_{j,\ell},s^*_{j,\ell})}|^{-\frac{1}{2}}}\right\}\times \\
&\sum_{j=1}^{m}\sum_{\ell=1}^{M^q}\Bigg\{\frac{K^{j,(r^*_{j,\ell},s^*_{j,\ell})}}{|\sigma^{j,(r^*_{j,\ell},s^*_{j,\ell})}|^{\frac{1}{2}}}\Bigg[\operatorname{inv}\big(\sigma^{j,(r^*_{j,\ell},s^*_{j,\ell})}\big) + \\
&\Big(\operatorname{inv}\big(\sigma^{j,(r^*_{j,\ell},s^*_{j,\ell})}\big)\delta^{j,(r^*_{j,\ell},s^*_{j,\ell})} - \widehat{x}_{k|k}\Big)\times \\
&\Big(\operatorname{inv}\big(\sigma^{j,(r^*_{j,\ell},s^*_{j,\ell})}\big)\delta^{j,(r^*_{j,\ell},s^*_{j,\ell})} - \widehat{x}_{k|k}\Big)'\Bigg]\Bigg\}.
\end{aligned}
\tag{30}
$$

- Update the matrices $\mathcal{A}_k$ and $\mathcal{B}_k$.

$$
\begin{aligned}
\mathcal{A}_k &= \begin{bmatrix} \alpha^{1,1}_{k|k} & \alpha^{1,2}_{k|k} & \dots & \alpha^{1,M^q}_{k|k} \\ \alpha^{2,1}_{k|k} & \alpha^{2,2}_{k|k} & \dots & \alpha^{2,M^q}_{k|k} \\ \vdots & \vdots & \vdots & \vdots \\ \alpha^{m,1}_{k|k} & \alpha^{m,2}_{k|k} & \dots & \alpha^{m,M^q}_{k|k} \end{bmatrix} \\
&\triangleq \Big[\operatorname{inv}\big(\sigma^{\gamma,r^*_{\gamma,\eta},s^*_{\gamma,\eta}}\big)\delta^{\gamma,(r^*_{\gamma,\eta},s^*_{\gamma,\eta})}\Big]_{\substack{1\le\gamma\le m\\1\le\eta\le M^q}}
\end{aligned}
\tag{31}
$$

and

$$
\begin{aligned}
\mathcal{B}_k &= \begin{bmatrix} \Sigma^{1,1}_{k|k} & \Sigma^{1,2}_{k|k} & \dots & \Sigma^{1,M^q}_{k|k} \\ \Sigma^{2,1}_{k|k} & \Sigma^{2,2}_{k|k} & \dots & \Sigma^{2,M^q}_{k|k} \\ \vdots & \vdots & \vdots & \vdots \\ \Sigma^{m,1}_{k|k} & \Sigma^{m,2}_{k|k} & \dots & \Sigma^{m,M^q}_{k|k} \end{bmatrix} \\
&\triangleq \Big[\operatorname{inv}\big(\sigma^{\gamma,r^*_{\gamma,\eta},s^*_{\gamma,\eta}}\big)\Big]_{\substack{1\le\gamma\le m\\1\le\eta\le M^q}}
\end{aligned}
\tag{32}
$$

- The un-normalised mode probability corresponding to the expectation $E\big[Z_k = \boldsymbol{e}_j \mid \mathcal{Y}_k\big]$, is approximated by

$$q^j_k \triangleq \frac{1}{(2\pi)^{d/2}\Phi(y_k)}\sum_{\ell=1}^{M^q}\zeta^{j,(r^*_{j,\ell},s^*_{j,\ell})}. \tag{33}$$

  To normalise the function $q^j_k(x)$, it is divided by sum of all terms $\{q^1_k,\dots,q^m_k\}$, for example

$$P(Z_k = \boldsymbol{e}_j \mid \mathcal{Y}_k) = \frac{\langle \boldsymbol{q}_k, \boldsymbol{e}_j\rangle}{\langle \boldsymbol{q}_k, \boldsymbol{1}\rangle}. \tag{34}$$

- For each $j \in \{1,2,\dots,m\}$ update the normalised weights $\{W^{(j,1)}_k, W^{(j,2)}_k,\dots,W^{(j,M^q)}_k\}$, with the equation

$$W^{(j,\ell)}_k \triangleq \frac{\zeta^{j,(r^*_{j,\ell},s^*_{j,\ell})}}{\sum_{j=1}^{m}\sum_{\ell=1}^{M^q}\zeta^{j,(r^*_{j,\ell},s^*_{j,\ell})}}. \tag{35}$$

Return to Step 1

Remark V.1 It is worth noting that the recursions given at (29) and (18) bear a striking resemblance to the form of the well know Kalman filter. This is not surprising. Suppose one knew the state of the Markov chain Z at each time k, consequently one would equivalently know the parameters of the dynamical system at this time. In such a scenario the Kalman filter in fact the optimal filter.

Remark V.2 The alogrithm presented in this article can be formulated with no hypothesis management, at the cost of accuracy, by setting $M^q = 1$.

VI. Nonlinear Systems

In many practical examples, inherently nonlinear stochastic hybrid systems can be successfully linearised and thereafter techniques such as those presented in this article may be routinely applied. Typical scenarios, are those with a hybrid collection of state dynamics, each observed through a common nonlinear mapping, or, alternatively, a hybrid collection of models, some linear, some nonlinear. One well known example of hybrid state dynamics observed through a nonlinear mapping is the bearings only tracking problem. In this case the hidden state process is measured through an inverse tan function and then corrupted by additive noise. The standard approach to such a problem is to first linearise the dynamics. However, to apply such techniques, a one-step-ahead prediction of the hidden state process is required. If the hybrid system being considered has either deterministic or constant parameters, this calculation is routine. By contrast, the same calculation for a stochastic hybrid system requires the joint prediction the future state and the future model. Extending the state estimator given at (29), we give the one-step-ahead predictor in the next Lemma.

Lemma 1 *Write*

$$\widehat{x}_{k|k}(j) \triangleq E\big[\langle Z_k, \boldsymbol{e}_j \rangle x_k \mid \mathcal{Y}_k\big]. \tag{36}$$

The quantity defined at (36), is a conditional mean estimate of x_k computed on those ω-sets which realise the event $Z_k = \boldsymbol{e}_j$. For the hybrid stochastic system defined by the dynamics at (3) and (4), the one-step prediction is computed by the following equation

$$\begin{aligned} \widehat{x}_{k+1|k} &\triangleq E[x_{k+1} \mid \mathcal{Y}_k] \\ &= \sum_{i=1}^{m} \sum_{j=1}^{m} \pi_{(j,i)} A_i \widehat{x}_{k|k}(j). \end{aligned} \tag{37}$$

Proof:

$$\begin{aligned} \widehat{x}_{k+1|k} = &\sum_{i=1}^{m} E\big[\langle Z_{k+1}, \boldsymbol{e}_j \rangle A_i x_k \mid \mathcal{Y}_k\big] + \\ &\qquad \sum_{i=1}^{m} E\big[\langle Z_{k+1}, \boldsymbol{e}_j \rangle B_i w_{k+1} \mid \mathcal{Y}_k\big] \\ = &\sum_{i=1}^{m} E\big[\langle \Pi Z_k, \boldsymbol{e}_j \rangle A_i x_k \mid \mathcal{Y}_k\big] + \\ &\qquad \sum_{i=1}^{m} E\big[\langle \Pi Z_k, \boldsymbol{e}_j \rangle B_i w_{k+1} \mid \mathcal{Y}_k\big] \\ = &\sum_{i=1}^{m} E\Big[\sum_{j=1}^{m} \langle Z_k, \boldsymbol{e}_j \rangle \langle \Pi Z_k, \boldsymbol{e}_j \rangle A_i x_k \mid \mathcal{Y}_k\Big] \\ = &\sum_{i=1}^{m} \sum_{j=1}^{m} \pi_{(j,i)} A_i E\big[\langle Z_k, \boldsymbol{e}_j \rangle x_k \mid \mathcal{Y}_k\big] \\ = &\sum_{i=1}^{m} \sum_{j=1}^{m} \pi_{(j,i)} A_i \widehat{x}_{k|k}(j) \end{aligned} \tag{38}$$

■

The $j = 1, 2, \ldots, m$ quantities $\hat{x}_{k|k}(j)$ may be computed as follows:

$$\widehat{x}_{k|k}(j) = \frac{\displaystyle\sum_{\ell=1}^{M^q} \left\{ \frac{K^{j,(r^*_{j,\ell}, s^*_{j,\ell})}}{|\sigma^{j,r^*_{j,\ell}, s^*_{j,\ell}}|^{\frac{1}{2}}} \operatorname{inv}\{\sigma^{j,r^*_{j,\ell}, s^*_{j,\ell}}\} \delta^{j,(r^*_{j,\ell}, s^*_{j,\ell})} \right\}}{\displaystyle\sum_{\ell=1}^{M^q} \left\{ K_{k,k-1}^{j,(r^*_{j,\ell}, s^*_{j,\ell})} |\sigma^{j,(r^*_{j,\ell}, s^*_{j,\ell})}|^{-\frac{1}{2}} \right\}}. \tag{39}$$

VII. Example

To demonstrate the performance of the algorithm described above, we consider a vector-valued state and observation process and a two-state Markov chain. Our Markov chain has a transition matrix

$$\Pi = \begin{bmatrix} 0.98 & 0.02 \\ 0.01 & 0.99 \end{bmatrix} \tag{40}$$

and an initial distribution $\big[0.4, 0.6\big]'$. The stochastic system model parameters $\big\{A, B, C, D\big\}$, for the two models considered, are each listed below.

$$A_1 = \begin{bmatrix} -0.8 & 0 \\ 0 & 0.2 \end{bmatrix}, \qquad A_2 = \begin{bmatrix} 0.8 & 0 \\ 0 & -0.2 \end{bmatrix}, \tag{41}$$

$$B_1 = B_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{42}$$

$$C_1 = \begin{bmatrix} 2 & 0.5 \\ 0.2 & 1 \end{bmatrix}, \qquad C_2 = \begin{bmatrix} 1 & 0.5 \\ 2 & 0.2 \end{bmatrix}, \tag{43}$$

$$D_1 = D_2 = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}. \tag{44}$$

A single realisation of this hybrid state and observation process was generated for 100 samples. Typical realisations of the estimated mode probability and the estimated state process are given, respectively, in Figures 1 and 2. The estimation of the hidden state process was compared against the exact filter, that is, the Kalman filter supplied with the exact parameter values A_k, B_k, C_k, D_k. This comparison is somewhat artificial as, one never has knowledge of the the hidden Markov chain in real problem settings, nonetheless, this example does serve to show the encouraging performance of the Gaussian mixture estimator.

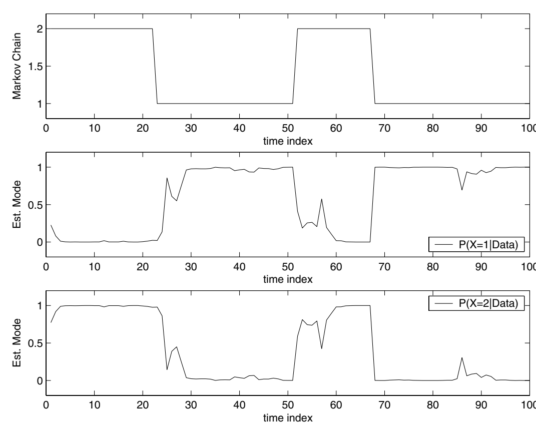

Fig. 1. Exact Markov chain and the estimated mode probability generated by the Gaussian mixture scheme.

References

[1] G. Ackerson and K. Fu, "On state estimation in switching environments", IEEE Transactions on Automatic Control 15(1), pp. 10-17, 1970.

[2] H. Blom, An efficient filter for abruptly changing systems, 23rd IEEE Conference on Decision and Control, Las Vegas USA, November 1984.

[3] R. J. Elliott, *Stochastic Calculus and Applications*, Springer Verlag, 1982.

[4] Elliott, R. J., Dufour, F. and Sworder, D., Exact hybrid filters in discrete time, IEEE Transactions on Automatic Control, 41, 1996, pp. 1807-1810.

[5] R. J. Elliott and W. P. Malcolm, Reproducing Gaussian Densities and Linear Gaussian Detection, Systems and Control Letters, 40 (2000), pp. 133-138.

[6] R. J. Elliott, L. Aggoun and J. B. Moore, *Hidden Markov Models Estimation and Control*, Springer Verlag Applications of Mathematics Series 29, 1995.

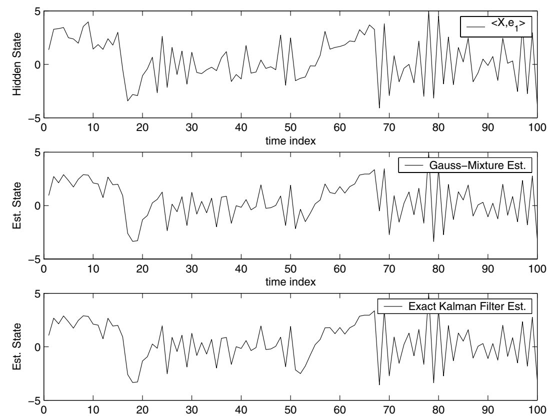

Fig. 2. True hidden state process and two estimates of this process. Here we plot only the first component of the state process. Subplot 2 is the state estimate process generated from the exact Kalman filter. Subplot 3 (the lowest subplot) is the state estimate process generated by the Gaussian mixture filter.

[7] L. Aggoun and R. J. Elliott, *Measure Theory and Filtering*, Cambridge University Press, 2004.

[8] R. J. Elliott, F. Dufour and W. P. Malcolm, State and Mode Estimation for Discrete-Time Jump Markov Systems, SIAM Journal of Optimisation and Control, *to appear*.

[9] H. W. Sorenson and Alspach, D. L.,Recursive Bayesian Estimation Using Gaussian Sums, Automatica, Volume 7, Number 4, July 1971, pp. 465-479.

[10] D. Sworder, R. Vojak and R. Hutchins, Gain adaptive Tracking, Journal of Guidance, Control and Dynamics, 16(5), pp. 865-873, 1993.

[11] J. K. Tugnait, Adaptive Estimation and Identification for Discrete Systems with Markov Jump Parameters, IEEE Transactions on Automatic Control, Volume AC-27, Number 5, October 1982.

[12] A. J. Viterbi, Error Bounds for the White Gaussian and Other Very Noisy Memoryless Channels with Generalized Decision Regions, IEEE Transactions on Information Theory, Volume IT, Number 2, March 1969.