Distributed Bayesian Fault diagnosis in Collaborative Wireless Sensor Networks.

Hichem Snoussi
ISTIT/M2S, University of Technology of Troyes,
12, rue Marie Curie, 10000, France
Email: snoussi@utt.fr

Cédric Richard
ISTIT/M2S, University of Technology of Troyes,
12, rue Marie Curie, 10000, France
Email: richard@utt.fr

***Abstract*— In this contribution, we propose an efficient collaborative strategy for online change detection, in a distributed sensor network. The collaborative strategy ensures the efficiency and the robustness of the data processing, while limiting the required communication bandwith. The observed systems are assumed to have each a finite set of states, including the abrupt change behavior. For each discrete state, an observed system is assumed to evolve according to a linear state-space model. An efficient Rao-Blackwellized collaborative particle filter (RB-CPF) is proposed to estimate the a posteriori probability of the discrete states of the observed systems. The Rao-Blackwellization procedure combines a sequential Monte Carlo filter with a bank of distributed Kalman filters. Only sufficient statistics are communicated between smart nodes. The spatio-temporal selection of the leader node and its collaborators is based on a trade-off between error propagation, communication constraints and information content complementarity of distributed data.**

I. Introduction

In this paper, the signal processing objective is to online detect the state change of a system observed by a sensor network. The efficient online state detection, in an automatic way, is very important for the system functioning security. In fact, according to each state, the system should adopt a specific behavior. For example, an autonomous robot must be able to detect its state and carry out repairs if necessary, without human intervention, by processing the data received from the on-board sensors [1], [2]. One can also mention the use of the sensor networks for the monitoring of production systems in order to face the industrial risks, the monitoring of the houses for safety or the house automation, the air and transport control in general, intelligent alarms for the prevention of natural disasters. With such systems, the automatic control of an event or an incident rests on the reliability of the network for a an efficient and robust decision-making.

For the above purpose, collaborative information processing in sensor networks is becoming a very attractive field of research. In such a sensor network, the sensors role is not limited to detect and transmit the data to a central unit where they are processed. Individual sensors have the capability to process the data and transmit only pertinent information to a fusion unit. The sensors have the ability to collaborate, exchange information to ensure an optimal decision. Such sensors are called smart sensors or smart nodes. Contrary to the centralized approach, the system does not depend on a unique processing unit whose damaging leads to the entire system failure. Every smart sensor is able to play a central role and provide a suboptimal decision. The system is thus very robust against a probable foreign attack or a technical failure of the central unit. In addition, as collected data are locally processed, only pertinent information is exchanged between smart nodes, limiting hence the required channel communication bandwidth. In fact, in a centralized network, all sensors transmit raw data to a unique processing unit, increasing the required communication bandwidth.

Concerning the data processing at each smart node and the fusion rule, we adopt a probabilistic approach to model the system dynamics. The system is described by a jump Markov linear Gaussian model where the conditional Gaussians depend on the discrete state of the system and also on the sensor. The state change detection is resumed in the posterior marginal probability of the discrete state. To solve the inference problem, we use the particle filter as an approximate Monte Carlo inference method able to deal with the intractable analytical aspect of the dynamical system update. Our contribution consists in proposing and implementing a collaborative distributed particle filter for estimating the marginal a posteriori probabilities of the system discrete states. Recently, distributed particle filters were proposed in literature [3], [4]. In the previously proposed distributed particle filters, the conditional distributions of the distributed collected data (likelihoods) are assumed to be independent. Therefore, applying these particle filters to the jump Markov models, one needs to consider jointly the continuous and the discrete states of the system. As shown in [1], in a centralized processing, the particle filtering of the joint state leads to poor results. Our contribution consists thus in extending the Rao-Blackwellized approach, proposed in [1], in a distributed environment. The leader node collaborates with the remaining nodes at each time step. The temporal selection of the leader node is based on a trade-off between information relevance, communication cost and propagation error. The spatial selection of the leader collaborators relies on the same trade-off except that the information relevance takes an information complementarity form. The main difficulty of the spatial collaboration, within the Rao-Blackwellized distributed particle filter, is the fact that the sensors marginal likelihoods are no more independent. We show in the proposed collaborative strategy how to circumvent this difficulty while propagating only sufficient second order

statistics through the sensor network.

The paper is organized as follows: in Section II, the probabilistic change detection model within the optimal centralized particle filter are briefly described. The section III contains the two main contributions of this paper: (i) an optimal online change detection procedure resulting from the spatial collaboration between the leader node and its collaborators, (ii) an information theoretic based criteria for the spatio-temporal selection of the leader node and its collaborators, under communications constraints. In section IV, numerical results, corroborating the proposed algorithm effectiveness, are shown.

II. Centralized online change detection

In this section, we briefly recall the particle filter method for online change detection. It is an approximate Monte Carlo method estimating, recursively in time, the posterior probabilities of the discrete state of the system, given the observations. Moreover, the particle filter provides a point mass approximation of the distributions of the hidden continuous states. For more details and a comprehensive review of the particle filter see [5].

A. Distributed State Space Model

The Bayesian change detection algorithm is based on a discrete time jump Markov linear state-space model. This model involves two different hidden states: a discrete state and a continuous state. The discrete state changes in time according to a first order Markov model. For each discrete state, the system, observed by a sensor network composed of M nodes, evolves in time according to a different linear Gaussian model:

$$\left\{\begin{array}{lcl} z_t & \sim & P(z_t \mid z_{t-1}) \\ \boldsymbol{x}_t & = & \boldsymbol{A}(z_t)\boldsymbol{x}_{t-1} + \boldsymbol{B}(z_t)\boldsymbol{w}_t \\ \boldsymbol{y}_t^{(m)} & = & \boldsymbol{C}_m(z_t)\boldsymbol{x}_t + \boldsymbol{D}_m(z_t)\boldsymbol{v}_t^m, \;\; m = 1..M, \end{array}\right. \quad (1)$$

where $\boldsymbol{y}_t^{(m)} \in \mathbb{R}^{n_y}$ denotes the observations transmitted from the sensor C^m at time t to the central processing unit, $\boldsymbol{x}_t \in \mathbb{R}^{n_x}$ denotes the unknown continuous state and $z_t \in \mathcal{Z} = \{1..K\}$ denotes the unknown discrete state. The transition probability $P(z_t \mid z_{t-1})$ represents the prior information about the dynamic variation of the system. The noises $\boldsymbol{w}_t$ and $\boldsymbol{v}_t^m$ are distributed according to i.i.d Gaussians $\mathcal{N}(0, \boldsymbol{I}_{n_x})$ and $\mathcal{N}(0, \boldsymbol{I}_{n_y})$ respectively. Note that the hidden states and their stochastic *a priori* models do not depend on the sensor node as they are characteristic of the observed system dynamics. The model parameters $\{\boldsymbol{A}, \boldsymbol{B}, \{\boldsymbol{C}_m\}_{m=1}^M, \{\boldsymbol{D}_m\}_{m=1}^M\}$ are assumed to be known.

In this paper, we assume that, given the states $\boldsymbol{x}_t$ and z_t, the sensor noises are stochastically independent:

$$p(\boldsymbol{y}_t^{(1)}, ..., \boldsymbol{y}_t^{(M)} \mid \boldsymbol{x}_t, z_t) = \prod_{m=1}^{M} p_m(\boldsymbol{y}_t^{(m)} \mid \boldsymbol{x}_t, z_t).$$

Consequently, concatenating the observations gathered in the central unit, $\boldsymbol{y}_t = [\boldsymbol{y}_t^{(1)}, ..., \boldsymbol{y}_t^{(M)}]$, and replacing the distribution product $\prod p_m$ by an observation distribution p_y, the stochastic model (1) is rewritten as:

$$\left\{\begin{array}{lcl} z_t & \sim & P(z_t \mid z_{t-1}) \\ \boldsymbol{x}_t & = & \boldsymbol{A}(z_t)\boldsymbol{x}_{t-1} + \boldsymbol{B}(z_t)\boldsymbol{w}_t \\ \boldsymbol{y}_t & \sim & \mathcal{N}(\boldsymbol{C}(z_t)\boldsymbol{x}_t, \boldsymbol{R}_y(z_t)), \end{array}\right. \quad (2)$$

where $\boldsymbol{C} = [\boldsymbol{C}_1^T, ..., \boldsymbol{C}_M^T]^T$ and $\boldsymbol{R}_y$ is the block diagonal covariance matrix with block matrices equal to $\boldsymbol{D}_m\boldsymbol{D}_m^T$. Hence, the centralized processing relies on the usual jump Markov state space model.

The Bayesian online change detection is based on the estimation of the posterior marginal probability $P(z_t \mid \boldsymbol{y}_{1:t})$. However, the probabilistic system model (2) involves hidden continuous variables $\boldsymbol{x}_{0:t}$. Therefore, the computation of the marginal distribution involves two intractable integrals: integration with respect to the past of the discrete time Markov chain $z_{0:t-1}$ and integration with respect to the hidden continuous states $\boldsymbol{x}_{0:t}$:

$$P(z_t \mid \boldsymbol{y}_{1:t}) = \sum_{z_{0:t-1}} \int p(z_{0:t}, \boldsymbol{x}_{0:t} \mid \boldsymbol{y}_{1:t}) d\,\boldsymbol{x}_{0:t}$$

Therefore, one has to resort to Monte Carlo approximation where the joint posterior distribution $p(z_{0:t}, \boldsymbol{x}_{0:t} \mid \boldsymbol{y}_{1:t})$ is approximated by the point-mass distribution of a set of weighted samples (called particles) $\{z_{0:t}^{(i)}, \boldsymbol{x}_{0:t}^{(i)}, w_t^{(i)}\}_{i=1}^N$:

$$\hat{P}_N(z_{0:t}, \boldsymbol{x}_{0:t} \mid \boldsymbol{y}_{1:t}) = \sum_{i=1}^{N} w_t^{(i)} \delta_{z_{0:t}^{(i)}, \boldsymbol{x}_{0:t}^{(i)}}(d\,\boldsymbol{x}_{0:t}, z_{0:t}),$$

where $\delta_{z_{0:t}^{(i)}, \boldsymbol{x}_{0:t}^{(i)}}(d\,\boldsymbol{x}_{0:t}, z_{0:t})$ denotes the Dirac function.

Based on the same set of particles, the marginal posterior probability (of interest) $P(z_t \mid \boldsymbol{y}_{1:t})$ can also be approximated as follows:

$$P(z_t = k \mid \boldsymbol{y}_{1:t}) \simeq \sum_{i=1}^{N} w_t^{(i)} \mathcal{I}(z_t^{(i)} = k),$$

where $\mathcal{I}(.)$ denotes the indicator function.

In the Bayesian importance sampling (IS) method, the particles $\{z_{0:t}^{(i)}, \boldsymbol{x}_{0:t}^{(i)}\}_{i=1}^N$ are sampled according to a proposal distribution $\pi(z_{0:t}, \boldsymbol{x}_{0:t} \mid \boldsymbol{y}_{1:t})$ and $\{w_t^{(i)}\}$ are the corresponding normalized importance weights:

$$w_t^{(i)} \propto \frac{p(\boldsymbol{y}_{1:t} \mid z_{0:t}^{(i)}, \boldsymbol{x}_{0:t}^{(i)}) p(z_{0:t}^{(i)}, \boldsymbol{x}_{0:t}^{(i)})}{\pi(z_{0:t}^{(i)}, \boldsymbol{x}_{0:t}^{(i)} \mid \boldsymbol{y}_{1:t})}.$$

B. Sequential Monte Carlo

Sequential Monte Carlo (SMC) consists of propagating the trajectories $\{z_{0:t}^{(i)}, \boldsymbol{x}_{0:t}^{(i)}\}_{i=1}^N$ in time without modifying the past simulated particles. The normalized importance weights are then recursively computed in time as:

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{p(\boldsymbol{y}_t \mid z_t^{(i)}, \boldsymbol{x}_t^{(i)}) p(z_t^{(i)}, \boldsymbol{x}_t^{(i)} \mid z_{0:t-1}^{(i)}, \boldsymbol{x}_{0:t-1}^{(i)})}{\pi(z_t^{(i)}, \boldsymbol{x}_t^{(i)} \mid z_{0:t-1}^{(i)}, \boldsymbol{x}_{0:t-1}^{(i)}, \boldsymbol{y}_{1:t})}. \quad (3)$$

For the considered jump Markov linear state-space model (2), one can adopt the transition prior as the proposal distribution:

$$\pi(z_t^{(i)}, \boldsymbol{x}_t^{(i)} | z_{0:t-1}^{(i)}, \boldsymbol{x}_{0:t-1}^{(i)}, \boldsymbol{y}_{1:t}) = p_x(\boldsymbol{x}_t | \boldsymbol{x}_{t-1}, z_t) P(z_t | z_{t-1}).$$

in which case the weights are updated according to the likelihood function:

$$w_t^{(i)} \propto w_{t-1}^{(i)}\, p(\boldsymbol{y}_t \mid z_t^{(i)}, \boldsymbol{x}_t^{(i)}). \tag{4}$$

C. Rao-Blackwellized SMC

Considering the joint state $\{\boldsymbol{x}_t, z_t\}$, the SMC algorithm yields poor online detection results. An efficient Rao-Blackwellized SMC, proposed in [1], considerably improves the state estimation. The principle of this procedure consists in noting that given the discrete state, the continuous state is *a posteriori* Gaussian. Thus, based on a bank of Kalman filters, one can sequentially update the marginal *a posteriori* probability $p(z_t \mid \boldsymbol{y}_{1:t})$. In fact, the probability of the trajectory $z_{0:t}$ satisfies the following recursion:

$$p(z_{0:t} | \boldsymbol{y}_{1:t}) = p(z_{0:t-1} | \boldsymbol{y}_{1:t-1}) \frac{p(\boldsymbol{y}_t | \boldsymbol{y}_{1:t-1}, z_{0:t}) P(z_t | z_{t-1})}{p(\boldsymbol{y}_t \mid \boldsymbol{y}_{1:t-1})}$$

In the SMC algorithm, predicting the discrete states $\{z_t^{(i)}\}$ according to the transition prior $P(z_t | z_{t-1})$ leads to the following particle weight updating:

$$w_t^{(i)} \propto w_{t-1}^{(i)}\, p(\boldsymbol{y}_t \mid \boldsymbol{y}_{1:t-1}, z_{0:t}^{(i)}) \tag{5}$$

The computation of the Gaussian data prediction distribution $p(\boldsymbol{y}_t | \boldsymbol{y}_{1:t-1}, z_{0:t}^{(i)})$ is based on the mean $\boldsymbol{y}_{t|t-1} = \mathrm{E}\big[\boldsymbol{y}_t | \boldsymbol{y}_{1:t-1}\big]$ and covariance $\boldsymbol{S}_t = cov(\boldsymbol{y}_t | \boldsymbol{y}_{1:t-1})$ online updates. These second order statistics are jointly updated with the mean and covariance of the continuous state by a Kalman filter as follows:

$$\begin{aligned}
\boldsymbol{\mu}_{t|t-1}^{(i)} &= \boldsymbol{A}(z_t^{(i)}) \boldsymbol{\mu}_{t-1|t-1}^{(i)} \\
\boldsymbol{\Sigma}_{t|t-1}^{(i)} &= \boldsymbol{A}(z_t^{(i)}) \boldsymbol{\Sigma}_{t-1|t-1}^{(i)} \boldsymbol{A}(z_t^{(i)})^T + \boldsymbol{B}(z_t^{(i)}) \boldsymbol{B}(z_t^{(i)})^T \\
\boldsymbol{S}_t^{(i)} &= \boldsymbol{C}(z_t^{(i)}) \boldsymbol{\Sigma}_{t|t-1}^{(i)} \boldsymbol{C}(z_t^{(i)})^T + \boldsymbol{R}_y(z_t^{(i)}) \\
\boldsymbol{y}_{t|t-1}^{(i)} &= \boldsymbol{C}(z_t^{(i)}) \boldsymbol{\mu}_{t|t-1}^{(i)} \\
\boldsymbol{\mu}_{t|t}^{(i)} &= \boldsymbol{\mu}_{t|t-1}^{(i)} + \boldsymbol{\Sigma}_{t|t-1}^{(i)} \boldsymbol{C}(z_t^{(i)})^T \boldsymbol{S}_t^{-1(i)} (\boldsymbol{y}_t - \boldsymbol{y}_{t|t-1}^{(i)}) \\
\boldsymbol{\Sigma}_{t|t}^{(i)} &= \boldsymbol{\Sigma}_{t|t-1}^{(i)} - \boldsymbol{\Sigma}_{t|t-1}^{(i)} \boldsymbol{C}(z_t^{(i)})^T \boldsymbol{S}_t^{-1(i)} \boldsymbol{C}(z_t^{(i)}) \boldsymbol{\Sigma}_{t|t-1}^{(i)}
\end{aligned}$$

where $\boldsymbol{\mu}_{t|t-1} = \mathrm{E}\big[\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1}\big]$, $\boldsymbol{\Sigma}_{t|t-1} = cov(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1})$, $\boldsymbol{\mu}_{t|t} = \mathrm{E}\big[\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}\big]$ and $\boldsymbol{\Sigma}_{t|t} = cov(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t})$. The predictive density is then simply evaluated by:

$$p(\boldsymbol{y}_t \mid \boldsymbol{y}_{1:t-1}, z_{0:t}^{(i)}) = \mathcal{N}(\boldsymbol{y}_t\,;\, \boldsymbol{y}_{t|t-1}, \boldsymbol{S}_t)$$

The centralized Rao-Blackwellized SMC algorithm is summarized in Figure 1.

Sequential sampling step:
- For $i = 1, ..., N$, sample from the transition prior:

$$\hat{z}_t^{(i)} \quad \sim \quad P(z_t \mid z_{t-1}^{(i)})$$

Weight updating step:
-For $i = 1, ..., N$, update the sufficient statistics (jointly with the Kalman filter) and evaluate the importance weights:

$$w_t^{(i)} \quad \propto \quad p(\boldsymbol{y}_t \mid \boldsymbol{y}_{1:t-1}, z_{0:t}^{(i)})$$

Resampling step:
- Select with replacement from $\{\hat{z}_{0:t}^{(i)}\}_{i=1}^N$ with probabilities $\{w_t^{(i)}\}$ to obtain N particles $\{z_{0:t}^{(i)}\}_{i=1}^N$

Fig. 1. Centralized Rao-Blackwellised particular filter algorithm

III. Collaborative online change detection

In a sensor network, each node must be able to treat the received data, to make a local decision and to communicate it in an autonomous way with the close nodes to which it is connected. This co-operation is intended to ensure best decision-making possible in spite of the limits in terms of power consumption and processing capability. The purpose of this work is to propose an efficient collaborative distributed version of the Rao-Blackwellized particle filter. In the following, we describe the proposed collaborative strategy.

A. temporal leader node selection

The temporal collaboration consists in selecting, after the sequential probability update, the leader node at the next time step. The selection procedure is based on ranking the nodes according to an information-theoretic cost function $J(m)$. The first ranked node m^* ($\arg\max_m J(m)$) is the next leader candidate. At time step $t-1$, the chosen cost function is a trade-off between information gain and compression loss:

$$J_t(m) = \mathcal{I}(m) + \alpha \mathcal{E}(m) \tag{6}$$

where the first term of the above criteria represents the information content relevance of the measured data on the node m, at the time step t:

$$\mathcal{I}(m) = \mathrm{E}\big[D_{KL}(p(\boldsymbol{y}_t^m \mid \boldsymbol{x}_t, z_t) \,||\, p(\boldsymbol{y}_t^m \mid \boldsymbol{y}_{1:t-1}, z_{0:t}))\big] \tag{7}$$

where D_{KL} is the Kullback-Leibler divergence between the likelihood and the data predicted density, the expectation is evaluated according to the joint filtering distribution $p(\boldsymbol{x}_t, z_{0:t} \mid \boldsymbol{y}_{1:t-1})$. This can be considered as a data augmentation version of criteria proposed in [6] for sensor management. The second term $\mathcal{E}(m)$ is the message error when transferring sufficient statistics from the leader node $m^*(t)$ to node m under the communication constraint $c_m < c_{max}$, where c_m is the communication cost of transferring information to node m. The negative coefficient α represents the trade off between the information gain and compression loss.

1) Computation of the information gain: In [6], a Monte Carlo procedure is proposed to compute the first term of the cost function (6). However, in our problem setting, using the jump Markov linear state model, the term $\mathcal{I}$ can be evaluated with a Rao-Blackwellized scheme. In fact, given the discrete state trajectory $z_{0:t}^{(i)}$, the likelihood $p(\boldsymbol{y}_t^m \mid \boldsymbol{x}_t)$ and the predictive distribution $p(\boldsymbol{y}_t^m \mid \boldsymbol{y}_{1:t-1}, z_{0:t}^{(i)})$ are both Gaussians and the expectation of the Kullback-Leibler divergence[1] in expression (7) can be exactly evaluated as follows:

$$\begin{aligned} \underset{\mid z_{0:t}^{(i)}}{\mathcal{I}}(m) &= \tfrac{1}{2}\log|\boldsymbol{I}_m + (\boldsymbol{D}_m(z_t)\boldsymbol{D}_m(z_t)^T)^{-1} \\ &\qquad \boldsymbol{C}_m(z_t)\boldsymbol{\Sigma}_{t|t-1}^{(i)}\boldsymbol{C}_m(z_t)^T| \end{aligned}$$

where the subscript "$_{z_{0:t}^{(i)}}$" means that the expectation is evaluated conditioned on the discrete state, $\boldsymbol{I}_m$ denotes the identity matrix and $\boldsymbol{\Sigma}_{t|t-1}^{(i)}$ is the predicted covariance $\boldsymbol{A}(z_t^{(i)})\boldsymbol{\Sigma}_{t-1|t-1}^{(i)}\boldsymbol{A}(z_t^{(i)})^T + \boldsymbol{B}(z_t^{(i)})\boldsymbol{B}(z_t^{(i)})^T$. It can be easily noted that maximizing the term $\mathcal{I}_{\mid z_{0:t}^{(i)}}(m)$ relies on the maximization of the information/noise ratio, where the information content is evaluated by the matrix $\boldsymbol{C}_m(z_t)\boldsymbol{\Sigma}_{t|t-1}^{(i)}\boldsymbol{C}_m(z_t)^T$ (norm of the observation matrix in the state covariance basis). The trajectory $z_{0:t}^{(i)}$ is composed of the particle past trajectory $z_{0:t-1}^{(i)}$ having $w_{t-1}^{(i)}$ as the importance weight and the predicted $z_t^{(i)}$ according to the transition prior $P(z_t|z_{t-1})$. The information criteria $\mathcal{I}(m)$ is thus approximated by a Monte Carlo scheme as follows:

$$\begin{aligned} \mathcal{I}(m) &= \mathrm{E}\big[\mathcal{I}_{\mid z_{0:t}}\big] = \sum_{z_{0:t}} \mathcal{I}_{\mid z_{0:t}} p(z_{0:t} \mid \boldsymbol{y}_{1:t-1}) \\ &\approx \sum_{z_{0:t}} \mathcal{I}_{\mid z_{0:t}} w_{t-1}^{(i)} \end{aligned}$$

2) Computation of the compression loss: Propagating all the particles $\{\boldsymbol{\mu}_{t|t}^{(i)}, \boldsymbol{\Sigma}_{t|t}^{(i)}, w_t^{(i)}\}$ is not allowed in a wireless sensor network because of the communications constraints. The KD-tree Gaussian mixture is a suitable approximation when communicating distribution messages [7]. The KD-tree is a multi-scale mixture of Gaussian approximation of a given data set. It consists in describing a large data set (particles) with a set a few sub-trees, each sub-tree is a Gaussian whose statistics can be recursively computed. The top node of the tree is the largest scale and the leaf nodes represent the finest scales. The internal nodes represent intermediate resolutions. See figure 2 for an illustration.

The set of Kalman means and covariances is approximated by a set of nodes $\mathcal{S}$ containing one and only one ancestor of each leaf node. Increasing the resolution of the KD-tree representation is simply done by replacing the nodes $s \in \mathcal{S}$ by their left and right children nodes. In order to control the error propagation, one needs a divergence measure between probability densities. Following the arguments in [7], the maximum log-error:

$$ML(p,q) = \max_x |\log p(x)/q(x)| \tag{8}$$

[1]The Kullback-Leibler divergence between two Gaussians $(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ is $\frac{1}{2}(\mathrm{tr}\left[\boldsymbol{\Sigma}_1\boldsymbol{\Sigma}_2^{-1}\right] - \log\boldsymbol{\Sigma}_1\boldsymbol{\Sigma}_2^{-1} - m + (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T\boldsymbol{\Sigma}_2^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2))$.

is very suitable for bounding the belief propagation error and also it is adapted to the KD-tree representation. Controlling the temporal propagation error while respecting the communication constraints consists in a trade-off between the resolution of the KD-tree representation and its encoding cost. As the resolution increases (going from top to bottom in the tree), the approximation error decreases while the communication cost increases. This can be easily implemented by recursively dividing the node $s \in \mathcal{S}$ having the maximum error measure while respecting the allowed communication cost.

Deciding the hand-over consists in comparing the information gain / compression loss ratio, computed for the selected leader candidate m_t^*, with a threshold β. In words, the hand-over to the node m_t^* is allowed if:

$$\frac{\mathcal{I}(m_t^*)}{\mathcal{I}(m_t^*) + \alpha\mathcal{E}(m_t^*)} > \beta$$

The threshold β is an increasing function of the energy reserve communicated by the active node's battery. If the energy reserve is very low ($\beta \approx 0$), the hand-over is almost surely done. However, if the energy reserve is at a correct level, the active node will take into consideration the information gain before performing the hand-over.

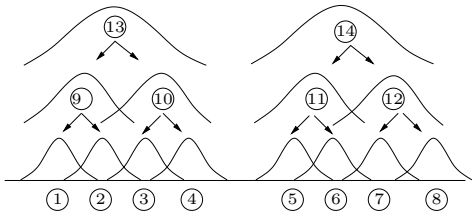

Fig. 2. KD-tree approximation of the Kalman mixture updates: Components 1 to 4 are the leaf nodes for the state $z_t = 1$ and components 5 to 8 are the leaf nodes for the state $z_t = 2$.

B. Spatial collaborative detection

An other important new feature of the proposed distributed Rao-Blackwellized particle filter is the spatial collaboration between the leader node and its selected collaborator nodes at each time step. The spatial collaboration is based on 2 alternating steps: (i) the selection of the collaborator nodes path with a recursive procedure ensuring the distributed data information complementarity and (ii) the spatial update of the particle weights, the particles being predicted in the leader node.

In the following, we outline the above two steps. For the clarity of presentation and notation convenience, $(\boldsymbol{\mu}_{t|t}^{(i,0)}, \boldsymbol{\Sigma}_{t|t}^{(i,0)})$ will denote the predicted Kalman mean and covariance $(\boldsymbol{\mu}_{t|t-1}^{(i)}, \boldsymbol{\Sigma}_{t|t-1}^{(i)})$, $\{w_t^{(i,0)}\}$ denotes their corresponding importance weight computed in the leader node C_0. The prediction is performed in the leader node C_0.

1) Particle weight updating: In this paragraph, we show how the weight of a predicted state is updated taking into account the data of the leader node and also the data collected by the collaborator nodes, under the communication

constraints. The communication constraints do not allow the propagation of raw data. Therefore, only sufficient statistics are exchanged between the leader node and its collaborators. The data measured at the leader node C_0 and its L collaborators $C_1, .., C_L$ are denoted $\{\boldsymbol{y}_t^0, \boldsymbol{y}_t^1, ..., \boldsymbol{y}_t^L\}$ respectively. Contrary to the previously proposed distributed particle filters in literature, in the jump Markov model, the likelihood of the discrete state $p(\boldsymbol{y}_t^0, \boldsymbol{y}_t^1, ..., \boldsymbol{y}_t^L \mid \boldsymbol{y}_{1:t-1}, z_{0:t})$ can not be factorized into $\prod_{l=0}^{L} p(\boldsymbol{y}_t^l \mid \boldsymbol{y}_{1:t-1}, z_{0:t})$. In fact, the predicted densities are dependent through the hidden continuous state. Consequently, the weight $w_t^{(i)} \propto p(\boldsymbol{y}_t^0, \boldsymbol{y}_t^1, ..., \boldsymbol{y}_t^L \mid \boldsymbol{y}_{1:t-1}, z_{0:t})$ of the predicted state $z_t^{(i)}$ can not be updated by a simple cumulative product. However, the computation of the complete likelihood can be performed with a Kalman filter procedure. In fact, the complete likelihood can be decomposed with the sequential Bayes' rule as follows:

$$p(\boldsymbol{y}_t^0, \boldsymbol{y}_t^1, ..., \boldsymbol{y}_t^L \mid \boldsymbol{y}_{1:t-1}, z_{0:t}) = p(\boldsymbol{y}_t^0 \mid \boldsymbol{y}_{1:t-1}, z_{0:t}) \times \prod_{l=1}^{L} p(\boldsymbol{y}_t^l \mid \boldsymbol{y}_t^{l-1}, ..., \boldsymbol{y}_t^0, \boldsymbol{y}_{1:t-1}, z_{0:t}) \quad (9)$$

The predicted density $p(\boldsymbol{y}_t^0 \mid \boldsymbol{y}_{1:t-1}, z_{0:t})$ in the product (9) is updated according to the usual Kalman filter based on the data $\boldsymbol{y}_t^0$. Similarly, the subsequent predictive data densities $p(\boldsymbol{y}_t^l \mid \boldsymbol{y}_t^{l-1}, ..., \boldsymbol{y}_t^0, \boldsymbol{y}_{1:t-1}, z_{0:t})$ are evaluated by a Kalman filter, where the predicted mean and covariance are the updated mean and covariance computed and sent by the node C_{l-1}. Thus, the main difference with an usual Kalman filter is the fact there is not a temporal prediction, the predicted statistics are the updated statistics by the previous collaborator node.

Figure 3 illustrates the collaborative updating of the Kalman means, covariances and particle weights, at each time step.

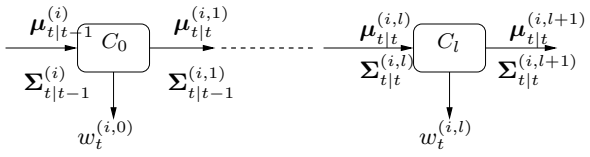

Fig. 3. Spatial Kalman update of the mean, covariance and particle weight.

Until now, we have considered the spatial update of one particle weight $w_t^{(i)}$. As we have mentioned in the previous section, updating all the particles is not possible under the communication constraints. Fortunately, the KD-tree approximation preserves the same structure of the Kalman mixture scheme. The computed means, covariances and weights of the KD-tree Gaussian mixture can be put in correspondence with the updated Kalman means $\boldsymbol{\mu}_{t|t}^{(i)}$, the updated Kalman covariances $\boldsymbol{\Sigma}_{t|t}^{(i)}$ and the particle weights $w_t^{(i)}$.

2) Recursive path selection: The selection of collaborator nodes can be performed in a recursive manner: each selected collaborator, after updating the particle weights, selects one and only one next collaborator. This recursion is necessary to ensure the information complementarity and avoid thus unnecessary redundant information. The selection is based on the same cost function (6) as in the temporal case, leading to similar expressions. Figure 4 illustrates the global spatio-temporal path of selected leader and auxiliary collaborator nodes.

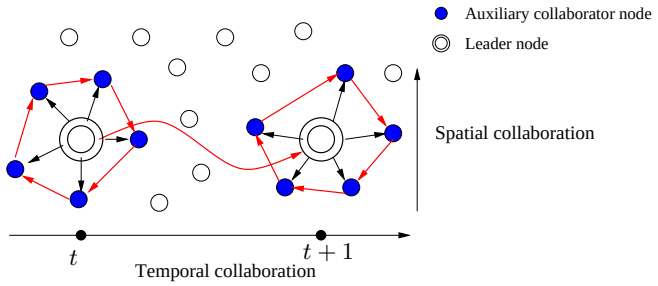

Fig. 4. Temporal leader selection + Recursive spatial collaborator path selection

IV. Numerical results

The proposed algorithm is applied on synthetic data generated according to the distributed jump Markov linear state space model (1). The system has 3 hidden discrete states ($K = 3$). The transition stochastic matrix is set as follows:

$$P(z_t \mid z_{t-1}) = \begin{pmatrix} 0.1 & 0.5 & 0.4 \\ 0.1 & 0.6 & 0.3 \\ 0.1 & 0.3 & 0.6 \end{pmatrix}$$

where the occurrence of the first state is lower the second and third states. The matrices $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}_m, \boldsymbol{D}_m)$ are set at random according to Gaussian distributions. The dimension of the hidden continuous state is set to $n_x = 2$ and the dimension of the observation is set to $n_y = 6$. The number of particles sequentially sampled at the leader nodes is $N = 100$. We have fixed severe communication constraints such that the maximum allowed collaborating nodes is 3 (leader node + 2 spatially collaborating nodes). Under these communication constraints, the resolution of the KD-tree approximation is only one Gaussian for each discrete state. In other words, the leader node communicates only 3 vector means and 3 covariances representing the Kalman mixture, to its spatially collaborating nodes.

Figure 5 shows the estimated *a posteriori* marginal discrete state probabilities $p(z_t \mid \boldsymbol{y}_{1:t})$. Note that, at each time step, the discrete states are not *a posteriori* equally distributed, avoiding ambiguity when estimating the states. In figure 6, the MAP estimate of the discrete states is plotted with the true discrete states. Note the accuracy of the proposed collaborative online detection, which is about 88%. The centralized Rao-blackwellized particle filter is also applied on the same set of data. Figure 7 shows the MAP discrete state estimates with the centralized processing whose classification precision is the same as the collaborative distributed algorithm (88%). This corroborates the efficiency of the proposed strategy under severe communication constraints. In order to further illustrate the effectiveness of the spatial collaboration strategy, figure 8 shows the detection performance of a distributed Rao-Blackwellized particle filter with only one leader node (no

collaborator nodes). Note that the performance has degraded to (68%).

V. Conclusion

We have proposed a distributed and collaborative version of the Rao-Blackwellized particle filter for online change detection. At each time step t, the selected leader node updates the posterior probability of the system discrete state. This update is based on a spatial collaboration with other nodes, called collaborator nodes. The nodes exchange only sufficient statistics (second order moments). The temporal selection of the leader node is based on a trade-off between information data *relevance* and compression loss under the communication constraints. Similarly, the spatial selection of collaborator nodes path is recursively designed and relies on a trade off between information *complementarity* and compression loss under the communication constraints.

In this work, we have assumed a jump Markov linear state space model for the observed system. The matrices involved in this model are assumed to be known (estimated in a training step). We are currently working on the extension to non linear models and the possibility to incorporate an unsupervised estimation of the model parameters.

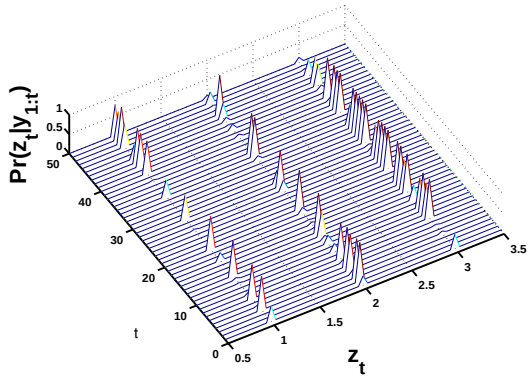

Fig. 5. A posteriori probabilities of the system discrete state

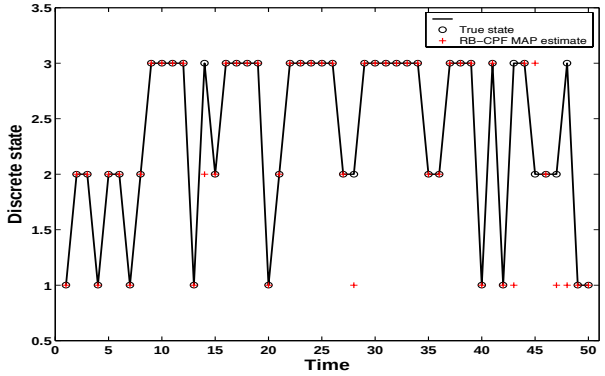

Fig. 6. Maximum a posteriori estimate of the system discrete state

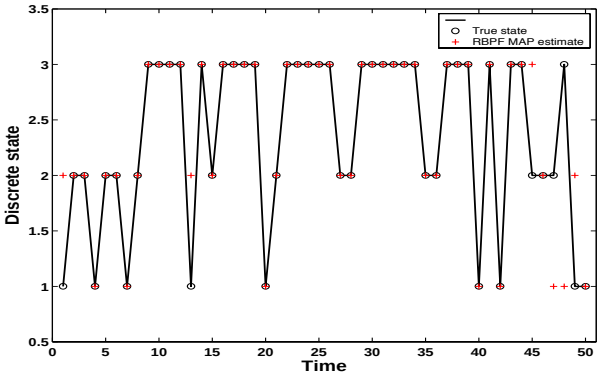

Fig. 7. Maximum a posteriori estimate of the system discrete state with a centralized processing

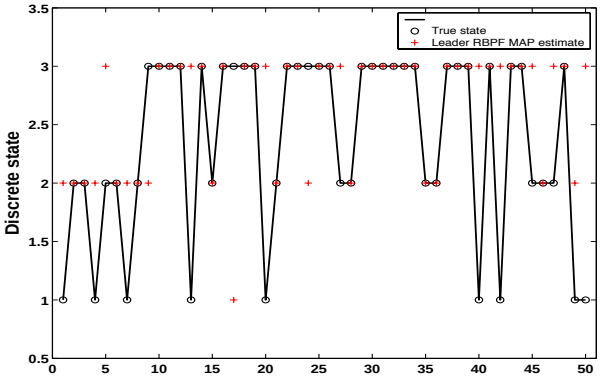

Fig. 8. Maximum a posteriori estimate of the system discrete state with only one leader node

References

[1] N. de Freitas, "Rao-Blackwellised Particle Filtering for Fault Diagnosis", in *IEEE Aerospace*, 2002.

[2] R. Washington, "On-board real-time state and fault identification for rovers", in *IEEE International Conference on Robotics and Automation*, 2000.

[3] A. Ihler, J. Fisher III, and A. Willsky, "Particle filtering under communications constraints", in *Proc. Statistical Signal Processing (SSP) 2005*, 2005.

[4] X. Sheng, Y. Hu, and P. Ramanathan, "Distributed particle filter with GMM approximation for multiple targets localization and tracking ion wireless sensor network", in *Information Processing in Sensor Networks, 2005. IPSN 2005*, April 2005, pp. 181–188.

[5] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering", *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.

[6] A. Doucet, B. Vo, C. Andrieu, and M. Davy, "Particle filtering for multitarget tracking and sensor management", in *Proc. Int. Conf. Inform. Fusion*, March 2002, vol. 1, pp. 474–481.

[7] A. Ihler, J. Fisher III, and A. Willsky, "Using sample-based representations under communications constraints", Tech. Rep. 2601, MIT, Laboratory for Information and Decision Systems, 2004.