

# INFORMATION-THEORETIC NODE SCHEDULING IN COLLABORATIVE SENSOR NETWORKS.

*Hichem Snoussi and Cédric Richard*

ICD/LM2S, University of Technology of Troyes,  
12, rue Marie Curie, 10000, France  
Email: snoussi@utt.fr

## ABSTRACT

In this contribution, we propose an efficient collaborative online filter in a wireless sensor network under communication constraints. The observed system is assumed to evolve according to a non linear state-space model. The distributed intelligent sensors collaborate to estimate the marginal probability of the hidden continuous state, within a Bayesian framework. The communication protocol relies on the activation of only few nodes. The selection of the leader node, through time, and its spatially collaborating nodes is based on a trade-off between data relevance (measured by an information criteria), message approximation error and communication constraints.

## 1. INTRODUCTION

A sensor network is a system made up of several tens to several hundreds of inter-connected nodes, made up, each one, of a sensor, an information processing unit and a communication block. The nodes have a zone of extremely reduced cover and are deployed in a dense way in heterogeneous environments. They are autonomous and have for that an energy reserve which renewal could be impossible, limiting thus their lifespan. 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 cooperation is intended to ensure best decision-making in spite of the limits in terms of power consumption and processing capability.

Recently, distributed particle filters were proposed in literature [1, 2]. In [2], a message approximating scheme based on a greedy KD-tree approximation is proposed. In [1], a full collaborative strategy is proposed with a GMM (Gaussian Mixture Model) message approximation. However, with the GMM approximation, the error propagation is not controlled and the model refinement is not straightforward as with the KD-tree approximation. Unlike the GMM modeling, the KD-tree Gaussian mixture modeling allows a more flexible control of the trade-off between the approximation precision and the communication constraints. In this contribution, we propose a full collaborative distributed particle filter based on the KD-tree approximation and a leadership selection protocol. In the proposed strategy, the spatio-temporal selection of the leader

node and its collaborators is based on a trade-off between the data information relevance, the communication constraint and the message approximation error.

The paper is organized as follows: in Section 2, the probabilistic optimal centralized particle filter is briefly described. The section 3 is the main contribution of this paper. We propose a collaborative particle filter in a distributed sensor network where the smart sensors exchange only density approximations (sufficient statistics) and select the leader nodes in an autonomous way. Finally, some simulation results corroborating the efficiency of the proposed algorithms are presented.

## 2. CENTRALIZED PARTICLE FILTERING

In this section, we briefly recall the particle filter method for filtering in nonlinear dynamical systems. It is an approximate Monte Carlo method estimating, recursively in time, the marginal posterior distribution of the continuous hidden state of the system, given the observations. The particle filter provides a point mass approximation of these distributions. For more details and a comprehensive review of the particle filter see [3].

The observed system evolves in time according to the following nonlinear dynamics:

$$\begin{cases} \mathbf{x}_t & \sim p_x(\mathbf{x}_t | \mathbf{x}_{t-1}, u_t) \\ \mathbf{y}_t^{(m)} & \sim p_m(\mathbf{y}_t^{(m)} | \mathbf{x}_t, u_t), \quad m = 1..M, \end{cases} \quad (1)$$

where  $\mathbf{y}_t^{(m)} \in \mathbb{R}^{n_y}$  denotes the observations transmitted from the sensor  $m$  at time  $t$ ,  $\mathbf{x}_t \in \mathbb{R}^{n_x}$  denotes the unknown continuous state, and  $u_t \in \mathcal{U}$  denotes a known control signal. The probability distribution  $p_x(\mathbf{x}_t | \mathbf{x}_{t-1}, u_t)$  models the stochastic transition dynamics of the hidden state. Given the state, the observations  $\mathbf{y}_t^{(m)}$  from each sensor  $m$  follow a stochastic model  $p_m(\mathbf{y}_t | \mathbf{x}_t, u_t)$ , where the stochastic aspect reflects the observation noise.

In this paper, we assume that, given the states  $\mathbf{x}_t$ , the sensor noises are stochastically independent.

The Bayesian filtering is based on the estimation of the posterior probability  $p(\mathbf{x}_t | \mathbf{y}_{1:t})$ . The nonlinear and the non Gaussian aspect of the transition distributions leads to intractable integrals when evaluating the marginals. Therefore, one has to resort to Monte Carlo approximation where

the joint posterior distribution  $p(\mathbf{x}_{0:t} | \mathbf{y}_{1:t})$  is approximated by the point-mass distribution of a set of weighted samples (called particles)  $\{\mathbf{x}_{0:t}^{(i)}, w_t^{(i)}\}_{i=1}^N$ :

$$\hat{p}_N(\mathbf{x}_{0:t} | \mathbf{y}_{1:t}) = \sum_{i=1}^N w_t^{(i)} \delta_{\mathbf{x}_{0:t}^{(i)}}(d\mathbf{x}_{0:t}),$$

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

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

$$w_t^{(i)} \propto \frac{p(\mathbf{y}_{1:t} | \mathbf{x}_{0:t}^{(i)})p(\mathbf{x}_{0:t}^{(i)})}{\pi(\mathbf{x}_{0:t}^{(i)} | \mathbf{y}_{1:t})}.$$

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

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{p(\mathbf{y}_t | \mathbf{x}_t^{(i)})p(\mathbf{x}_t^{(i)} | \mathbf{x}_{0:t-1}^{(i)})}{\pi(\mathbf{x}_t^{(i)} | \mathbf{x}_{0:t-1}^{(i)}, \mathbf{y}_{1:t})}. \quad (2)$$

### 3. COLLABORATIVE PARTICLE FILTERING

In the following, we propose a collaborative particle filter (CPF) where the smart nodes collaborate in sequentially updating the filtering distribution. They only exchange few statistics characterizing message approximations. The observed data  $\{\mathbf{y}_t^{(m)}\}_{m=1}^M$  are not propagated in the sensor network. The proposed distributed particle filter is characterized by a spatial and a temporal collaborative processing of the sequentially collected observations.

#### 3.1. Temporal collaborative processing

The temporal collaboration consists in selecting, after the sequential probability update, the leader node at the next time step. The selection procedure, performed by the leader node at time  $t-1$ , 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  $t$ . At time step  $t-1$ , the chosen cost function is a trade-off between information gain and compression loss:

$$J_t(m) = \mathcal{I}(m) + \alpha \mathcal{E}(m) \quad (3)$$

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

$$\mathcal{I}(m) = \mathbb{E}[D_{KL}(p(\mathbf{x}_t | \mathbf{y}_{1:t}) || p(\mathbf{x}_t | \mathbf{y}_{1:t-1}))] \quad (4)$$

Here,  $D_{KL}$  is the Kullback-Leibler divergence between the current and the predicted filtering distributions, the expectation is evaluated according to the predicted distribution  $p(\mathbf{y}_t | \mathbf{y}_{1:t-1})$  [4]. The second term  $\mathcal{E}(m)$  is the message error occurring when transferring the compressed filtering distribution from the leader node  $m^*(t-1)$  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  $\alpha$  represents the trade off between the information gain and compression loss. Note that  $\mathcal{E}(m^*(t-1)) = 0$ , meaning that the leader node may select itself as the next leader, when the increase of the data relevance of the other nodes does not compensate the compression loss.

The first term  $\mathcal{I}(m)$  can be evaluated with a Monte Carlo integration (see [4] for details). The second term  $\mathcal{E}(m)$  depends on the particle representation form exchanged between nodes. Propagating all the particles  $\{\mathbf{x}_t^{(i)}, w_t^{(i)}\}$  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 [5]. The KD-tree is multi-scale mixture of Gaussian approximation of a given data set. It consists in describing a large data set (particles) with a set of 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. The set of weighted particles  $\{\mathbf{x}^{(i)}, w^{(i)}\}$  can thus be approximated by a set of nodes  $\mathcal{S}$  containing one and only one ancestor of each leaf node:

$$\hat{p}(x_t | y_{1:t}) = \sum_{i=1}^N w^{(i)} \delta_{x^{(i)}}(x_t) \approx \sum_{s \in \mathcal{S}} \alpha_s \mathcal{N}(x_t; \boldsymbol{\mu}_s, \boldsymbol{\Sigma}_s) \quad (5)$$

where  $\mathcal{N}(\cdot)$  denotes the Gaussian density. Following the arguments in [5], the maximum log-error:

$$ML(p, q) = \max_x |\log p(x)/q(x)| \quad (6)$$

is very suitable for bounding the belief propagation error and also it is adapted to the KD-tree representation.

Deciding the hand-over consists in comparing the information gain / compression loss ratio, computed for the selected leader candidate  $m_t^*$ , with a threshold  $\beta$ . 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  $\beta$  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.

#### 3.2. Spatial collaborative processing

The basic difference between the temporal and the spatial collaboration is the following: Once the hand-off is performed between  $t-1$  and  $t$ , the leader node  $m^*(t-1)$  has no more control on the filtering distribution  $p(x_t | y_{1:t}^{(m_t^*)})$ . However, in the spatial case, the leader node can select its collaborating neighbors and then decide whether to take

into account this collaboration or not. In other words, after evaluating the collaboration, the leader node has the possibility to cancel the spatial update and transmit only its filtering distribution. The aim of the proposed procedure is to take into account the message error propagation in the network. If the ratio between the message error and the information gain becomes greater than a fixed threshold, the collaboration is useless and thus cancelled by the leader. In the following, we describe, in details, this collaboration protocol between the leader and the auxiliary nodes.

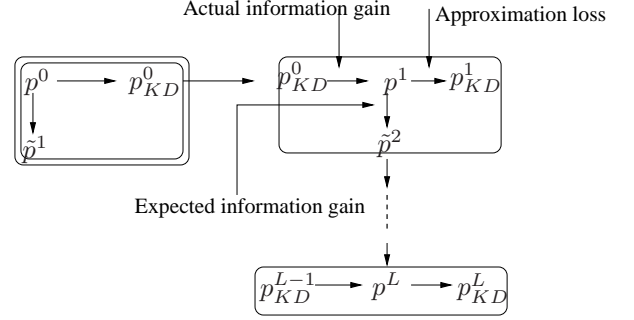
**1. Recursive auxiliary node selection:** As the auxiliary nodes are selected among the neighboring nodes, we assume that the communication cost is fixed. The number of the maximum number of the Gaussian components is therefore fixed to a value depending on the surface area where the auxiliary nodes are selected. The spatial collaborative updating of the filtering distribution, at the time step  $t$ , is iteratively performed. First, the leader node updates the filtering distribution  $p^0 = p(\mathbf{x}_t | \mathbf{y}_{1:t-1}, \mathbf{y}_t^0)$ , where the superscript  $0$  refers to the leader node. Then, the leader node selects one auxiliary node  $m^1$  if the ratio  $\mathcal{E}^0 / \mathcal{I}(m^1) < \gamma$ , where  $\mathcal{E}^0$  is the message error of the KD-tree compression,  $\mathcal{I}(m^1)$  (expression 4) the expected information gain and  $\gamma$  a fixed threshold. If the ratio  $\mathcal{E}^0 / \mathcal{I}(m^1)$  is greater than the threshold, then the leader node cancels the collaboration. When selected, the first auxiliary node  $m^1$  updates the filtering distribution to obtain  $p^1 = p(\mathbf{x}_t | \mathbf{y}_{1:t-1}, \mathbf{y}_t^0, \mathbf{y}_t^1)$ . Then, it computes the actual information gain as the Kullback-Leibler divergence  $\mathcal{I}_a(m^1)$  between the filtering distributions  $p_{KD}^0$  and  $p^1$ . If the ratio  $\mathcal{E}^0 / \mathcal{I}_a(m^1)$  is lower than the threshold  $\gamma$  then, the updated  $p^1$  is accepted, otherwise the leader node cancels the collaboration. This procedure can be considered as an *a posteriori* correction. It is useful when the computed expected information gain shows a significant difference with the actual information gain. If the collaboration is not cancelled, then the auxiliary node  $m^1$  selects the next auxiliary node  $m^2$ , based on the same procedure, and so on, until the maximum number  $L$  of auxiliary nodes is reached.

In order to analyze the divergence between  $p^0$  and  $p_{KD}^j$  and relate it to the information gain and error propagation, we need a divergence measure satisfying the triangle inequality. Unfortunately, the Kullback-Leibler divergence does not satisfy the triangle inequality in general. However, using the ML-measure and applying the triangle inequality to the successive error propagations and likelihood updates (see figure 1), the distance between  $p^0$  and  $p_{KD}^j$  (the KD-tree approximation of  $p^j$ ) can be bounded as follows:

$$\begin{cases} ML(p^0, p_{KD}^j) \leq \mathcal{I}_a^j + \mathcal{E}_p^j, \\ \mathcal{I}_a^j = 2 \sum_{l=1}^j ML(p_{KD}^{l-1}, p^l) \\ \mathcal{E}_p^j = 2 \sum_{l=0}^j ML(p^l, p_{KD}^l) \end{cases}$$

Thus, constraining the ratio  $\mathcal{E}^{j-1} / \mathcal{I}_a(m^j)$  to be lower

than the threshold  $\gamma$ , at each node selection step, ensures that the overall ratio is bounded by the same threshold  $\gamma$ .

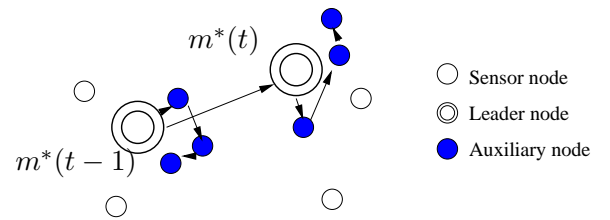


**Fig. 1.** Message passing and updating in the spatial collaboration step.

**2. Spatial filtering distribution updating:** The optimal centralized particle filtering requires that the sampled particles  $\{\mathbf{x}^{(i)}\}$  at the leader node must be weighted by the product of the likelihoods  $\prod_{l=1}^L p_{y_t}(\mathbf{y}_t^{(m_l)} | \mathbf{x}^{(i)})$ . However, to meet the communication constraints, only the KD-tree representations are exchanged between the collaborating nodes. We propose to update the filtering distribution in the following cumulative way:

- For  $l = 1..L$ , at the collaborating node  $m_t^l$ ,
- Sample  $N$  particles  $\{\tilde{\mathbf{x}}^{(l-1,i)}\}$  from  $p_{KD}^{l-1}$  sent by  $m_t^{l-1}$ ,
- Compute normalized weights  $w^i \propto p(\mathbf{y}_t^{(m_l)} | \tilde{\mathbf{x}}^{(l-1,i)})$ ,
- Resample the particles according to the normalized weights,
- Approximate the particle distribution  $p^l$  by a KD-tree  $p_{KD}^l$
- Send the KD-tree approximation  $p_{KD}^l$  to the next node  $m_t^{l+1}$ .

Using the above notation,  $p^0$  and  $p_{KD}^0$  represent the leader filtering distribution and its KD-tree approximation. Figure 2 illustrates the global spatio-temporal collaborative online Bayesian filter.



**Fig. 2.** Spatio-temporal belief propagation: The leader node and its collaborators are selected according to a trade-off between data information relevance and communication constraints, the spatial collaboration allows *a posteriori* correction of the filtering distribution.

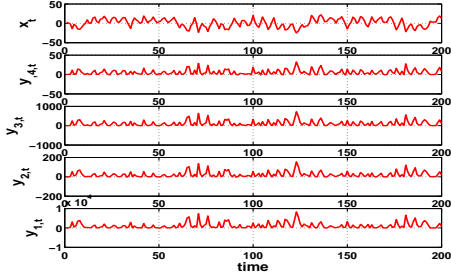
## 4. SIMULATION RESULTS

In order to illustrate the performances of the proposed collaborative particle filter, we consider the following non

linear reference model:

$$\begin{cases} x_t &= F(x_{t-1}, t, u(t)) + v_t \\ &= \frac{1}{2}x_{t-1} + 25x_{t-1}(1 + x_{t-1}^2)^{-1} + 8 \cos 1.2t + v_t. \\ y_{m,t} &= G_m(x_t, u(t)) + w_{m,t} \\ &= \beta_m x_t^2 + w_{m,t}, \quad m = 1..M \end{cases}$$

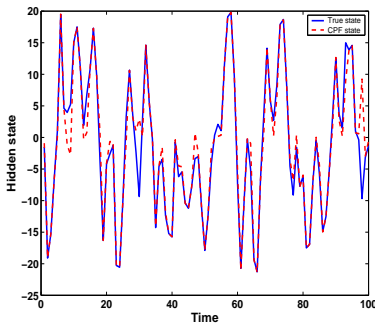
where  $v_t$  and  $\{w_{m,t}\}_{m=1}^M$  are mutually independent Gaussian noises with variances  $\sigma_v^2 = 10$  and  $\{\sigma_{w,m}^2\}_{m=1}^M$  randomly set between 1 and 2. Figure 3 illustrates the hidden state and observation time series data for 4 distributed sensors.



**Fig. 3.** From top to bottom: hidden state time series  $x_{1..T}$  and the 4 sensor time series observations  $\{y_{m,t}\}_{m=1}^4$ .

The collaborative particle filter is applied on this data set to estimate the marginal distribution  $p(x_t | y_{1:t})$ . We have used the transition prior as the importance sampling distribution (first collaborative strategy, see subsection 3.2) with  $N = 50$  particles. In figure 4, the estimated hidden state time series is plotted against the true corresponding time series, for a typical run. Note the ability of the collaborative filter to follow the hidden state dynamics. The hidden states are approximated by their empirical *a posteriori* expectation:

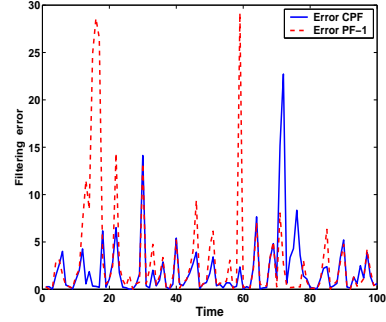
$$\hat{x}_t = \frac{1}{N} \sum_{i=1}^N x_t^{(i)}$$



**Fig. 4.** The empirical mean estimate of the hidden state time series.

In order to better illustrate the performances of the collaborative particle filter, we have plotted in figure 5 the residual estimation error of both the collaborative filter and the local particle filter  $PF_1$  updated by the leader node

$m^*$ . The local filter is obtained by storing the locally sampled particles without spatially collaborating with auxiliary nodes. Note that the collaborative filter clearly outperforms the local  $PF_1$  filter, except for a short time interval (around 70 s). In fact, it happens that the leader node is particularly very informative about the true state in this short interval for this experiment. However, the collaborative filter surpasses the local filter at each time step on the average (in a Monte Carlo sense) as the expected information content of the grouped nodes is higher than the local information content of the leader node.



**Fig. 5.** The estimation error of the CPF filter ( $|x_t - x_t^{CPF}|$ ) and the local particle filter ( $|x_t - x_t^{PF}|$ ).

## 5. CONCLUSION

In this contribution, a collaborative distributed Bayesian online filter is proposed. The collaboration is based on a spatio-temporal protocol to select only few active smart sensors. The selection of the collaborating nodes is based on a trade-off between the data information relevance (measured by an information theoretic criteria), the communication constraints and the propagation of the message approximation error.

## 6. REFERENCES

- [1] X. Sheng, Y. Hu, and P. Ramanathan, "Distributed particle filter with GMM approximation for multiple targets localization and tracking in wireless sensor network", in *Information Processing in Sensor Networks, 2005. IPSN 2005*, April 2005, pp. 181–188.
- [2] A. Ihler, J. Fisher III, and A. Willsky, "Particle filtering under communications constraints", in *Proc. Statistical Signal Processing (SSP) 2005*, 2005.
- [3] 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.
- [4] 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.
- [5] 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.