Sequential Variational Inference for Distributed Multi-Sensor Tracking and Fusion

Wei Du
Institut Montefiore, B28,
University of Liège
B-4000 Liège, Belgium
Email: weidu@montefiore.ulg.ac.be

Justus Piater
Institut Montefiore, B28,
University of Liège
B-4000 Liège, Belgium
Email: Justus.Piater@ulg.ac.be

***Abstract*—This paper presents a novel sequential variational inference algorithm for distributed multi-sensor tracking and fusion. The algorithm is based on a multi-sensor target representation where a target is represented jointly by its states at different sensors and a global state fusing all sensor data. A tree-structured graphical model is adopted to model the dependencies between these states at a time instant. In contrast to previous work, most of which are based on Belief Propagation, we propose an alternative variational inference algorithm, which combines importance sampling techniques and variational methods for graphical models, to infer the multi-sensor target states in time. In particular, the sequential variational inference algorithm distributes the global inference to each node in graphical models. With a message-passing scheme which is similar to the one in BP, the inference processes at different nodes collaborate so that each integrates information from all sensors. One contribution of this paper is the proper design of an importance function for generating samples to approximate the target distributions. Experiments on a synthetic example show that our method achieves comparable results with those by the BP-based methods.**

Keywords: belief propagation, particle fitler, variational inference, graphical model, importance sampling.

I. Introduction

In many applications such as environment monitoring and robotics, multiple distributed sensors are used to collect measurements in a scene. The measurements from these sensors have to be fused to extract useful information. A lot of systems have been developed based on either centralized or distributed architectures. It has been shown that centralized architectures have disadvantages of high communication and computation load at a single site and low survivability due to a single point of failure, thus distributed architectures are more preferable in practise [2], [4].

Commonly, a distributed architecture contains a set of sensor nodes, for collecting and processing measurements from each sensor, and a communication scheme for exchanging information between sensors. One advantage of this architecture is that it nicely incorporates graphical models [10] for modeling the dependencies between sensor data, which facilitates the use of a number of inference algorithms such as Belief Propagation (BP) [21]. Alternative to BP, variational methods have also proved powerful tools for solving inference problems on graphical models [10], [11].

In this paper, we present a novel sequential variational inference algorithm for distributed multi-sensor tracking and fusion. The algorithm is based on a multi-sensor target representation where a target is represented jointly by its states at different sensors and a global state fusing all sensor data. A graphical model is adopted to model the dependencies between these states. As the target states at different sensors are only connected to the global state, a tree-structured graphical model is formed. Thus, we distribute the inference of the multi-sensor target state at each node of the graphical model. In particular, the process at each sensor tracks the targets by processing local measurements. A central process fuses the local estimates collected from each sensor and sends the fusion results back to each sensor. Then, the local estimates at different sensors are updated by taking into account the information from other sensors. Although with a tree structure, the bidirectional connection between the central process and each sensor process allows the integration of the sensor data at all nodes of the graphical model. To have the inference processes at different nodes collaborate, a sequential variational inference algorithm is developed that combines the strength of both particle filters and variational methods for graphical models. Experiments on a synthetic example show that our method achieves good performance even in the presence of heavy level noise.

Given most related work on distributed inference in sensor networks are based on graphical models, BP is usually the first choice for solving the inference problem. For example, Paskin et al. and Cetin et al. proposed a similar algorithm for general inference in sensor networks based on Nonparametric Belief Propagation (NBP) [15], [1], [16], [9]. Ihler et al. and Kantas et al. used NBP for distributed sensor calibration or registration [8], [12]. In our previous work, we developed a Sequential Belief Propagation (SBP) algorithm for the inference in multi-sensor tracking and fusion [3]. As above-mentioned work used sampling based methods such as particle filters for approximating the target distributions, variational methods can also be used as an alternative. Jordon et al. described the use of variational methods for graphical models, which is considered a classic work in this field [11]. To the best of our knowledge, Vermaak et al. was the first to introduce variational methods in the field of tracking [18]. Hua et al. proposed an interesting method for solving inference problems on dynamic Markov models. The method combines importance sampling techniques with variational methods for graphical models, which is suitable for sequential inference tasks such as tracking [6], [7]. However, one of the important issues in Hua et al.'s method is the design of the importance function, which is not described clearly. In this paper, we adapt the sequential variational inference algorithm for the task of multi-sensor tracking and fusion. With a specifically-designed dynamic Markov model, we incorporate a suitable importance function for generating samples in state space. Somehow, the variational inference algorithm is similar to the BP-based inference methods in that they both pass messages between nodes. However, the messages in variational methods are different from those in BP. We tested our method on a synthetic example and achieved reliable results.

The rest of the paper is organized as follows. Section II briefly introduces variational methods for graphical models. Section III describes our method for distributed multi-sensor tracking and fusion, including the specifically designed graphical models and the details of the sequential variational inference algorithm. Results on synthetic examples are illustrated in Section IV.

II. Variational Methods for Graphical Models

Variational methods are well-developed techniques for finding extremal functions. We briefly describe the use of variational methods for graphical models. Consult [10], [11], [6], [7] for the details of variational methods and their applications on graphical models.

Generally, a graphical model consists of a set of state nodes $X = x_i, i = 1, \ldots, L$ and a set of measurement nodes $Z = z_i, i = 1, \ldots, L$. We assume each state node x_i is linked with a measurement node z_i and the directed link is associated with a likelihood function $p(z_i|x_i)$. If two states x_i and x_j are dependent on each other, an undirected link connects them and the link is associated with a potential function $\psi_{i,j}(x_i, x_j)$. Thus, the joint probability density function (pdf) corresponding to such a graphical model is

$$p(X, Z) \propto \prod_{(i,j)\in\varepsilon} \psi(x_i, x_j) \prod_{i\in\nu} p(z_i|x_i), \tag{1}$$

where ε is the set of undirected links between state nodes and ν is the set of directed links from state nodes to measurement nodes.

Our task is to infer the posterior $p(X|Z)$ which is in general intractable. Thus, we often infer each $p(x_i|Z)$, $i = 1, \ldots, L$, instead where BP has been widely used. Alternative to BP, variational methods can also be used to achieve an approximate solution. We assume a fully factorized approximation, i.e., we adopt the mean field approximation

$$p(X|Z) \approx Q(X) = \prod_i Q_i(x_i), \tag{2}$$

where $Q_i(x_i)$ is an independent approximation to $p(x_i|Z)$. Then we consider the well-known cost function

$$J(Q) = \log(p(Z)) - \mathrm{D_{KL}} \left(\prod_i Q_i(x_i) \parallel p(X|Z) \right), \tag{3}$$

where $\mathrm{D_{KL}}$ is the KL divergence or relative entropy between two distributions. To maximize $J(Q)$ is to minimize the KL divergence. Differentiating $J(Q)$ with respect to $Q_i(x_i)$ and setting it to 0, we have the following Euler-Largange equation,

$$\log Q_i(x_i) = E_Q\{\log p(X, Z)|x_i\} + \text{constant}, \tag{4}$$

where

$$E_Q\{\log p(X, Z)|x_i\} = \int \prod_{\{j\}\setminus i} Q_j(x_j) \log p(X, Z) \, \mathrm{d}\,\{x_j\} \setminus x_i \tag{5}$$

is the expectation of $\log p(X, Z)$ relative to $\prod_{\{j\}\setminus i} Q_j(x_j)$. $\{a\} \setminus b$ denotes all elements in $\{a\}$ except b. Substituting Eq. 1 and 2 into Eq. 4, we obtain the following fixed point equation,

$$\log Q_i(x_i) = \log p(z_i|x_i) + \sum_{j\in N(i)} \int Q_j(x_j) \log \psi(x_i, x_j) \, \mathrm{d}\, x_j + \text{constant}, \tag{6}$$

where $N(i)$ denotes the set of subscripts of the neighboring nodes of x_i.

Eq. 6 is perhaps the most important equation in variational inference. It can be seen that to calculate $Q_i(x_i)$, all $Q_j(x_j)$, $j \in N(i)$, have to be known. This implies the iterative calculation for each node in graphical models: calculate one while fixing the rest, and iterate until convergence. Another observation is that Eq. 6 has a similar form as the belief equation in BP, except that the message product in BP is replaced by the expectation in Eq. 5 in Variational inference. In a sense, the expectation in Eq. 5 can be regarded as the messages from neighboring nodes in variational inference.

III. Variational Inference for Multi-Sensor Tracking and Fusion

We now consider the variational inference in multi-sensor tracking and fusion. The problem is defined as targets of interest move in 3D and are captured by a few sensors separated by wide-baseline. Our task is to estimate the target states in 3D based on the noise-corrupted multi-sensor measurements. To avoid a centralized architecture, we distribute the global inference to each node in graphical models and perform a sequential variational inference algorithm.

A. Graphical Models

Suppose L sensors are used and each sensor collects one measurement for a target at each time instant. Denote the target state in 3D by $x_{t,0}$ and its states at different sensors by $x_{t,j}$, and let $z_{t,j}$ the measurement in sensor j at time t, $j = 1, \ldots, L$. Thus, we can define $\mathbf{X}_t = \{x_{t,1}, \ldots, x_{t,L}\}$ the multi-sensor target state at time t, $\mathbf{Z}_t = \{z_{t,1}, \ldots, z_{t,L}\}$ the multi-sensor measurement at time t, and $\mathbf{Z}^t = \{\mathbf{z}_1, \ldots, \mathbf{z}_t\}$ the multi-sensor measurements up to time t.

Given the above definitions, a graphical model is built, shown in Fig. 1(a), to model the dependencies between target states in 3D and at different sensors at time t. The model consists of three types of nodes: the central node associated with $x_{t,0}$, the sensor nodes associated with $x_{t,j}$, and the measurement nodes associated with $z_{t,j}$, $j = 1, \ldots, L$. We assume that $x_{t,j}$, $j = 1, \ldots, L$, are independent given $x_{t,0}$ so that a tree-structured model is formed. Note that the central node is not associated with a measurement and each sensor node is associated with the measurement captured by the sensor. Connecting the graphical models at different times results in a dynamic Markov model, shown in Fig. 1(b), which describes the evolution of the system in time. As $x_{t,j}$, $j = 1, \ldots, L$, are dependent on $x_{t,0}$, we add temporal links between $x_{t-1,0}$ and $x_{t,j}$, making the dynamic Markov model asymmetric. The reason for not adding links between $x_{t-1,j}$ and $x_{t,0}$ to make a fully coupled model is to reduce the model complexity. We will show that the adding of these temporal links is beneficial in the design of the importance function in the next section.

In both models in Fig. 1, the undirected link between $x_{t,j}$, $j = 1, \ldots, L$, and $x_{t,0}$ describes the mutual influence between each sensor node and the central node, and is associated with a potential function $\psi^t_{0,j}(x_{t,0}, x_{t,j})$. The directed link from $x_{t,j}$ to $z_{t,j}$, $j = 1, \ldots, L$, represents the measurement process and is associated with a likelihood function $p_j(z_{t,j}|x_{t,j})$. In Fig. 1(b), the directed links from $x_{t-1,i}$ to $x_{t,i}$, $i = 0, \ldots, L$, and from $x_{t-1,0}$ to $x_{t,j}$, $j = 1, \ldots, L$ represent the state transition processes and are associated with motion models $p(x_{t,i}|x_{t-1,i})$ and $p(x_{t,j}|x_{t-1,0})$ respectively.

According to Bayes' rule, the recursive inference of $p(X_t|Z^t)$ is formulated as

$$P(X_t|Z^t) \propto P(Z_t|X_t) \int P(X_t|X_{t-1})\, P(X_{t-1}|Z^{t-1})\, \mathrm{d}\, X_{t-1} \tag{7}$$

However, direct inference according to Eq. 7 is often intractable. Thus, we approximate each $p(x_{t,i}|Z^t)$ by a variational distribution $Q_{t,i}(x_{t,i})$, $i = 0, \ldots, L$, as shown below.

B. Sequential Variational Inference

To perform sequential inference, the temporal information has to be taken into account. Based on the dynamic Markov model in Fig. 1(b), we have

$$\begin{aligned} p(X_t, Z^t) \quad \propto \quad & \prod_{j=1}^{L} \psi(x_{t,0}, x_{t,j}) \prod_{j=1}^{L} p(z_j|x_j) \times \\ & \int p(X_t|X_{t-1}) p(X_{t-1}|Z^{t-1})\, \mathrm{d}\, X_{t-1}. \end{aligned} \tag{8}$$

Again, we adopt the mean field approximations

$$p(X_{t-1}|Z_{t-1}) \quad \approx \quad Q(X_{t-1}) = \prod_{i=0}^{L} Q_{t-1,i}(x_{t-1,i}), \tag{9}$$

$$p(X_t|Z_t) \quad \approx \quad Q(X_t) = \prod_{i=0}^{L} Q_{t,i}(x_{t,i}). \tag{10}$$

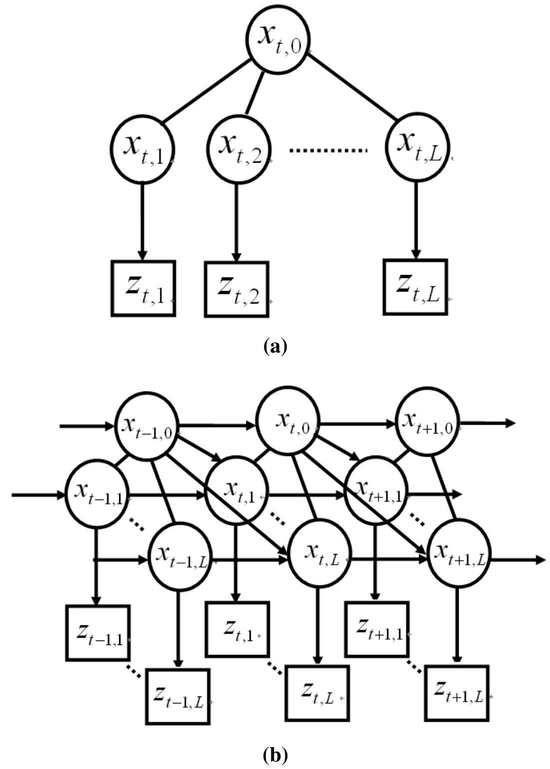

Figure 1. (a) Tree-structured graphical model for modeling the dependencies at time t. (b) Dynamic Markov model for the evolution of the system in time.

Then, given $Q_{t-1}(X_{t-1}) = \prod_i Q_{t-1,i}(x_{t-1,i})$ which approximates $p(X_{t-1}|Z^{t-1})$, we need to find $Q_t(X_t) = \prod_i Q_{t,i}(x_{t,i})$ to approximate $p(X_t|Z_t)$. Similar to the steps in Section II, we can construct the cost function

$$J_t(Q_t) = \log(p(Z^t)) - \mathrm{D_{KL}} \left(\prod_{i=0}^{L} Q_{t,i}(x_{t,i}) \parallel p(X_t|Z^t) \right). \tag{11}$$

Assuming independent motion models for each node, we have

$$p(X_t|X_{t-1}) = \prod_{i=0}^{L} p(x_{t,i}|x_{t-1,i}) \prod_{j=1}^{L} p(x_{t,j}|x_{t-1,0}). \tag{12}$$

Embedding Eqs. 8, 9, 10, and 12 into 11, and differentiating $J_t(Q_t)$ with respect to $Q_{t,i}(x_{t,i})$ and setting it to 0, we obtain the following fixed point equations for sensor nodes

$$\begin{aligned} \log Q_{t,j}(x_{t,j}) \quad = \quad & \log p(z_{t,j}|x_{t,j}) \\ & + \log \int p(x_{t,j}|x_{t-1,j}) Q_{t-1,j}\, \mathrm{d}\, x_{t-1,j} \\ & + \log \int p(x_{t,j}|x_{t-1,0}) Q_{t-1,0}\, \mathrm{d}\, x_{t-1,0} \\ & + \int Q_{t,0}(x_{t,0}) \log \psi_{0,j}(x_{t,0}, x_{t,j})\, \mathrm{d}\, x_{t,j} \\ & + \text{constant}, \end{aligned} \tag{13}$$

where $j = 1, \ldots, L$, and for the central node

$$\log Q_{t,0}(x_{t,0}) = \log \int p(x_{t,0}|x_{t-1,0}) Q_{t-1,0} \, \mathrm{d}\, x_{t-1,0} + \sum_{j=1}^{L} \int Q_{t,j}(x_{t,j}) \log \psi_{0,j}(x_{t,0}, x_{t,j}) \, \mathrm{d}\, x_{t,j} + \text{constant}. \quad (14)$$

Comparing Eqs. 13 and 14 with Eq. 6, we can see the only difference is the extra terms that model the state transition priors. As stated in Section II, we infer each $x_{t,i}$, $i = 0, \ldots, L$, iteratively using Eqs. 13 and 14 until convergence.

C. Monte Carlo Implementation

As is in Hua et al. [6], the sequential variational inference algorithm is implemented with Monte-Carlo techniques, i.e., each variational distribution $Q_{t,i}(x_{t,i})$ is represented by N weighted samples,

$$Q_{t,i}(x_{t,i}) \sim \{s_{t,j}^n, \pi_{t,j}^n\}_{n=1}^N, \; i = 0, \ldots, L.$$

As the samples are generated by importance sampling techniques, one issue is the design of the importance function. Hua et al. [5] used the state transition function $p(x_{t,i}|x_{t-1,i})$ for each $x_{t,i}$ in similar SBP algorithms. Simple and very often effective as it is, the problem is that the importance sampling of each $x_{t,i}$ from $p(x_{t,i}|x_{t-1,i})$ is independent for $i = 0, \ldots, L$. As the samples are supposed to approximate $Q_{t,i}(x_{t,i})$ which, according to Eqs. 13 and 14, contains information from neighboring nodes, the independent sampling may cause problems when e.g. a sensor is subject to heavy level noise such as outliers. To overcome this problem, Hua et al. [6] used the pre-learnt potential function $\psi_{i,j}(x_{t,i}, x_{t,j})$ as the importance function, which may not be suitable here because in our graphical models, all sensor nodes are linked to the central node, it is not clear how to sample $x_{t,0}$ from all $\psi_{0,j}(x_{t,0}, x_{t,j})$, $j = 1, \ldots, L$.

Note that in our dynamic Markov model in Fig. 1(b), for each sensor node $x_{t,j}$), there is an extra temporal link from $x_{t-1,0}$ besides that from $x_{t-1,j}$, $j = 1, \ldots, L$. This enables us to develop a simple layered importance sampling technique. For the central node which fuses sensor data, we use the simplest state transition function $p(x_{t,0}|x_{t-1,0})$ as the importance function. For the sensor nodes which may be subject to outliers, we generate samples from both $p(x_{t,j}|x_{t-1,j})$ and $p(x_{t,j}|x_{t-1,0})$, i.e., αN samples are generated from $p(x_{t,j}|x_{t-1,j})$ and the rest $(1-\alpha)N$ samples are generated from $p(x_{t,j}|x_{t-1,0})$. α is a trade-off between two transition functions and is set to 50% in our experiments. This is beneficial in maintaining consistency between the inference processes at different nodes. The Monte-Carlo implementation of the algorithm is given in Algorithm III-C.

Algorithm 1 Sequential Variational Inference for Distributed Multi-Sensor Tracking and Fusion

Input: $Q_{t-1,i}(x_{t-1,i}) \sim \{s_{t-1,i}^n, \pi_{t-1,i}^n\}_{n=1}^N$, $i = 0, \ldots, L$
Output: $Q_{t,i}(x_{t,i}) \sim \{s_{t,i}^n, \pi_{t,i}^n\}_{n=1}^N$, $i = 0, \ldots, L$

Re-sampling: Get $\{s_{t-1,i}^n, \frac{1}{N}\}_{n=1}^N$ from $\{s_{t-1,i}^n, \pi_{t-1,i}^n\}_{n=1}^N$, $i = 0, \ldots, L$.

$k = 0$

Prediction: Get $\{s_{t,0,k}^n\}_{n=1}^N$ from $p(x_{t,0}|x_{t-1,0})$ and $\{s_{t-1,i,k}^n\}_{n=1}^N$ from both $p(x_{t,j}|x_{t-1,j})$ and $p(x_{t,j}|x_{t-1,0})$.

Re-weighting: For $n = 1, \ldots, N$, let $\pi_{t,0,k}^n = \frac{1}{N}$ and $\pi_{t,j,k}^n = p(z_{t,i}|s_{t,j,k}^n)$, $j = 1, \ldots, L$.

repeat

$k = k + 1$.

Importance Sampling: Get $\{s_{t,0,k}^n\}_{n=1}^N$ from $\{s_{t,0,k-1}^n, \pi_{t,0,k-1}^n\}_{n=1}^N$ and $\{s_{t,0,k}^n\}_{n=1}^N$ from both $\{s_{t,j,k-1}^n, \pi_{t,j,k-1}^n\}_{n=1}^N$ and $p(x_{t,j}|x_{t-1,0})$, $j = 1, \ldots, L$.

Expectation Calculation: For $n = 1, \ldots, N$,

$$E_{x_{t,0}}(s_{t,0,k}^n) = \sum_{j=1}^{L} \sum_{m=1}^{N} \pi_{t,j,k-1}^m \log \psi_{0,j}(s_{t,0,k}^n, s_{t,j,k-1}^m),$$
$$E_{x_{t,j}}(s_{t,j,k}^n) = \sum_{m=1}^{N} \pi_{t,0,k-1}^m \log \psi_{0,j}(s_{t,0,k-1}^m, s_{t,j,k}^n), \quad j = 1, \ldots, L.$$

Re-weighting: For $n = 1, \ldots, N$,

$$\pi_{t,0,k}^n = \pi_{t,0,k-1}^n \exp(E_{x_{t,0}}(s_{t,0,k}^n)),$$
$$\pi_{t,j,k}^n = \pi_{t,j,k-1}^n p(z_{t,j}|s_{t,j,k}^n) \exp(E_{x_{t,j}}(s_{t,j,k}^n)), \quad j = 1, \ldots, L.$$

Normalization: $\pi_{t,i,k}^n = \pi_{t,i,k}^n / \sum_{m=1}^N \pi_{t,i,k}^m$, $i = 0, \ldots, L$, $n = 1, \ldots, N$.

until convergent or $k > 5$

Estimation: For $i = 0, \ldots, L$, let $s_{t,i}^{(n)} = s_{t,i,k}^{(n)}$ and $\pi_{t,i}^{(n)} = \pi_{t,i,k}^{(n)}$, then

$$Q_{t,i}(x_{t,i}) \sim \{s_{t,i}^{(n)}, \pi_{t,i}^{(n)}\}_{n=1}^N, \text{ and } \hat{x}_{t,i} = \sum_{n=1}^{N} s_{t,i}^{(n)} \pi_{t,i}^{(n)}.$$

D. Performance Analysis

It can be see that the time complexity of the inference process at the central node is $O(LN^2)$, where L is the number of the sensors and N is the number of samples, and that at each sensor node is $O(N^2)$. It seems that the central-node tracker is the bottleneck of the whole system. However, note that the central node processes no measurements. Indeed, Eqs. 13 and 14 shows clearly that the central node calculates the expectation relative to all $Q_{t,j}(x_{t,j})$, $j = 1, \ldots, L$, which is similar to the message product in BP, while each sensor node calculates the product of the local likelihood and the expectation relative to $Q_{t,0}(x_{t,0})$. For most systems with a reasonably small number of sensors, the time consumption at the central node is comparable with that at each sensor node. Thus, the algorithm is very suitable for parallel implementation. For those systems with a large number of sensors, a hierarchical architecture is preferable.

One important issue in distributed multi-sensor tracking and fusion is that the measurements from different sensors are often inconsistent, which causes the disagreement among the inference processes at different nodes. It has been shown that BP solves this inconsistency effectively due to the asymmetric property of the message passing scheme [3], [17]. Likewise, the sequential variational inference algorithm also maintains the consistency among the distributed inference processes with a similar "message-passing" scheme, which assures the inference at each node is performed based on the information from all sensors. Thus, the final decision at each node is always made based on the consensus.

IV. Results on a Synthetic Example

To best demonstrate the strengths of our sequential variational inference algorithm, we present experimental results on a simulated, multi-sensor tracking scenario. The problem considered here is a point target moving along a circular trajectory in a plane. Noise-corrupted observations in four different sensors are synthesized by projecting the point into each sensor. To make the problem challenging, 10% of the true measurements in each sensor are removed to simulate occlusions, and outliers are added to the four sensors but at different times to simulate distractor objects. In particular, the outliers are around the true measurements at time 40 in Sensor 1, at time 60 in Sensor 2, at time 70 in Sensor 3, and at time 80 in Sensor 4 respectively. This synthetic example is representative of the problems that are difficult to solve using classical multi-sensor fusion techniques, as the outliers are not distinguishable from the true measurements.

Figure 2 presents the tracking result of our algorithm in a typical simulation. It can be seen that the measurements missing in some sensors are compensated by other sensors. The true modes are correctly selected among the multi-modal distributions introduced by the outliers in each sensor.

We compared our algorithm to the BP-based methods. In particular, we have developed a SBP algorithm for the same multi-sensor tracking and fusion task [3]. The results of the SBP algorithm on the same data is shown in Fig. 3. Both methods succeed in all simulations with heavy level noise. The error between the estimates and the ground truth of both methods is shown in Fig. 4. It can be seen that our algorithm achieves reliable results that are comparable with those by the BP-based methods.

We also compared our algorithm to other particle-filter-based multi-sensor tracking algorithms in [14], [19]. To overcome the problem of inconsistent measurements, both contributions suggested to assess the tracking performance of each sensor. In particular, Wang et al. [19] evaluated each particle with the data in the most reliable sensor, which is adopted to do the comparison in this paper.

We argue that this tracking-performance-assessment strategy is sufficient only to deal with missing observations but not with distractors, as the distractors in one sensor may produce larger likelihoods than the true observations in other sensors. A comparison of our algorithm, the SBP algorithm and the tracking-performance assessment method by Wang et al. is shown in Fig. 5. It can be seen that the outliers pose serious problems to Wang et al.'s method, as it almost instantly fails when the outliers appear in one sensor but not in other sensors.

V. Conclusion and Future Work

We present a novel sequential variational inference algorithm for distributed multi-sensor tracking and fusion. Alternative to BP, which has been widely used for distributed inference on graphical models, our algorithm approximate the target distributions by fully factorized variational (mean field) distributions. Note that the fully factorized approximation does not ignore the dependencies among the random variables [20]. The use of variational methods for graphical models is beneficial in that with a similar message passing scheme as in BP, each local process integrates information from all sensors, resulting in collaborative inference of the multi-sensor target states.

Further study is expected on the comparison of BP and variational methods for graphical models. One potential advantage of variational methods is perhaps the proper design of a variational importance function. However, it has been shown by David Mackay [13] that the variational approximation is in general more compact than the true distribution. As heavy-tailed distributions are desirable for importance functions, the variational distributions obtained by variational methods is not a suitable choice for generating samples. Vermaak et al. [18] proposed an interesting approach to incorporating temporal priors in the variational approximation to the target distribution, which may be used in our future work.

References

[1] M. Cetin, L. Chen, J. Fisher, A. Ihler, R. Moses, M. J. Wainwright, and A. S. Willsky. Distributed fusion in sensor networks. *IEEE Signal Processing Magazine*, 42(7), July, 2006.

[2] C.-Y. Chong and S. Mori. Graphical models for nonlinear distributed estimation. In *the Seventh International Conference on Information Fusion*, pages 206–213, Stockholm, Sweden, 2004.

[3] W. Du and J. Piater. Data fusion by belief propagation for multi-camera tracking. In *the Nineth International Conference on Information Fusion*, 2006.

[4] A. Gad, M. Farooq, J. Serdula, and D. Peters. Multitarget tracking in a multisensor multiplatform environment. In *the Seventh International Conference on Information Fusion*, pages 206–213, Stockholm, Sweden, 2004.

[5] G. Hua and Y. Wu. Multi-scale visual tracking by sequential belief propagation. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 826–833, Washington, DC, 2004.

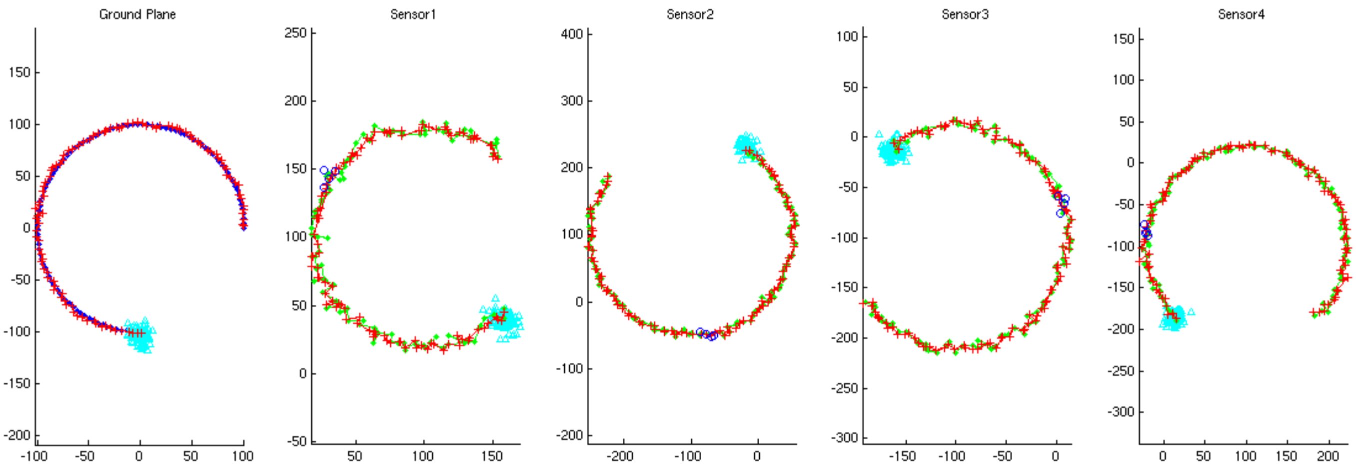

Figure 2. Tracking results of our sequential variational inference algorithm in one simulation. In the Ground-Plane view, the blue trajectory gives the ground truth. In Sensor 1–4, the green dots are the true, noise-corrupted observations, and the blue circles are the outliers. Note that the outliers are close to the true observations in different views at different times. The red trajectory in both the Ground-Plane view and Sensor 1–4 gives the estimates of the positions. The cyan triangles in Sensor 1–4 represent the particles at the final time instant.

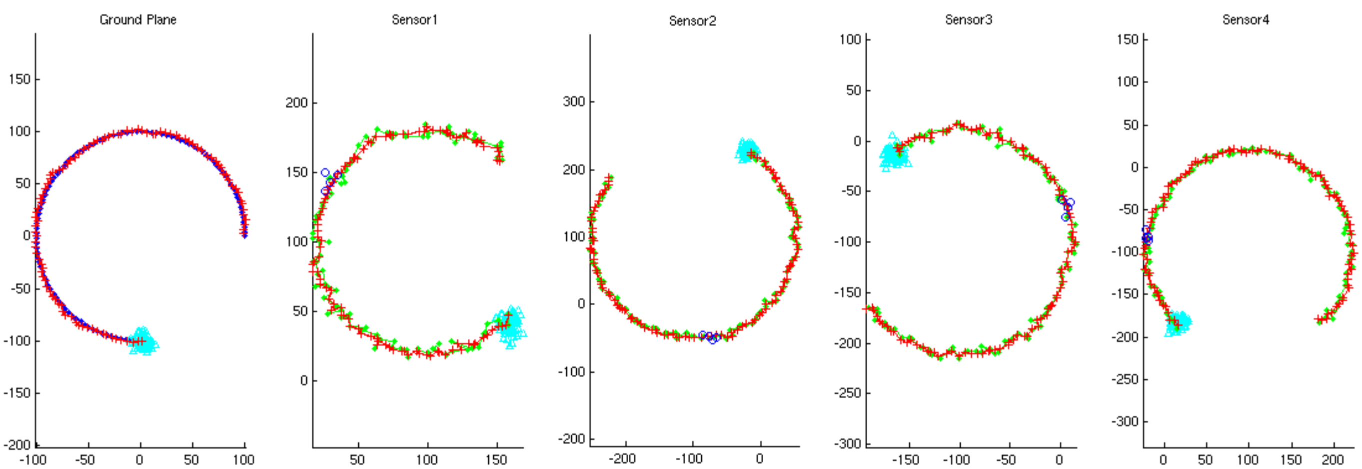

Figure 3. Tracking results of the SBP algorithm on the same data as above.

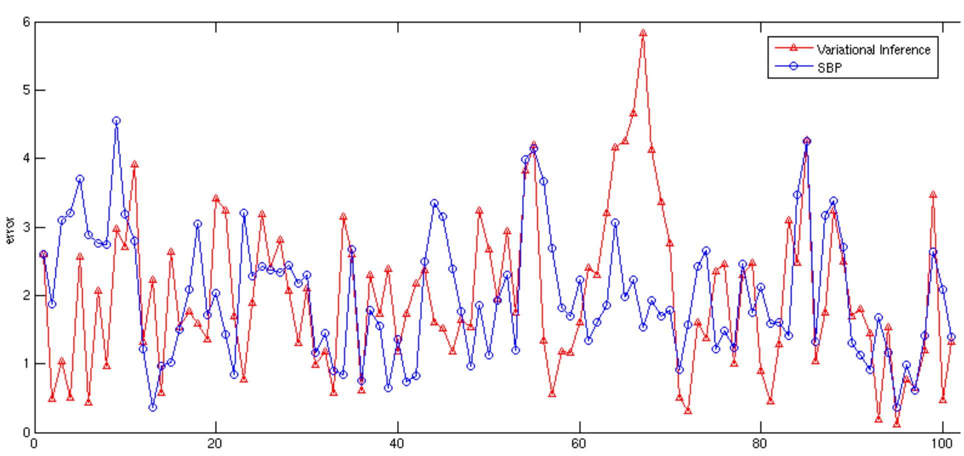

Figure 4. Error between the estimates in the ground plane and the ground truth of the two methods.

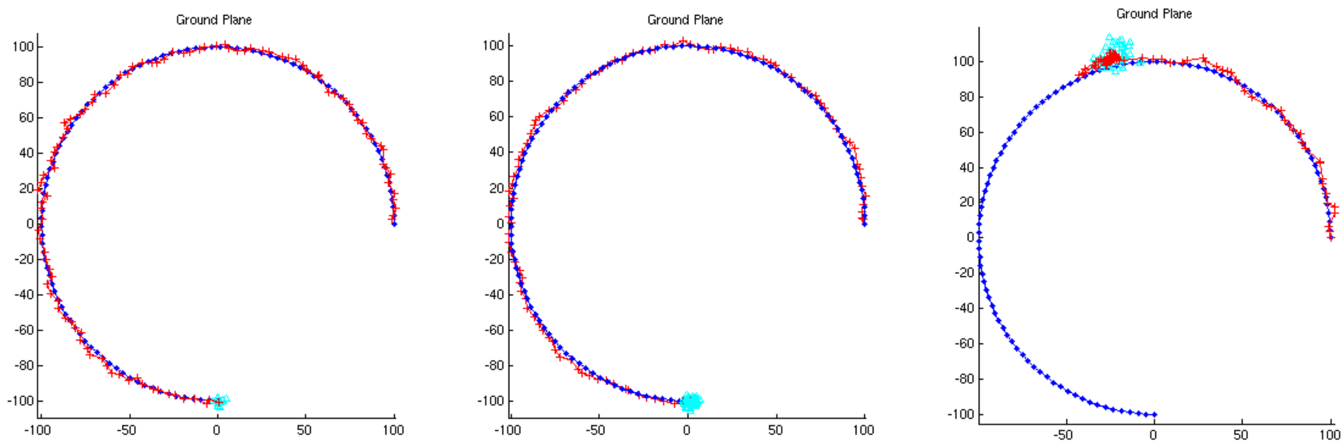

Figure 5. Comparison of the three tracking algorithms. From left to right are our sequential variational inference algorithm, the SBP algorithm, and the method by Wang et al. [19]. The comparison is based on the same data.

[6] G. Hua and Y. Wu. Sequential mean field variational analysis of structured deformable shapes. *Computer Vision and Image Understanding*, 101:87–99, 2006.

[7] G. Hua and Y. Wu. Variational maximum a posteriori by annealed mean field analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(11):1–15, 2006.

[8] A. Ihler, J. Fisher, R. Moses, and A. Willsky. Nonparametric belief propagation for self-calibration in sensor networks. In *the third international symposium on Information processing in sensor networks*, Berkeley, 2004.

[9] M. Isard. Pampas: Real-valued graphical models for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 613–620, Madison, WI, 2003.

[10] M. Jordan. *Learning in Graphical Models*. MIT Press, Cambridge, MA, 1999.

[11] M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, 1999.

[12] N. Kantas, S. Singh, and A. Doucet. A distributed recursive maximum likelihood implementation for sensor registration. In *the Nineth International Conference on Information Fusion*, 2006.

[13] D. Mackay. *Information theory, Inference, and learning algorithms*. Cambridge University Press, 2003.

[14] K. Nummiaro, E. Koller-Meier, T. Svoboda, D. Roth, and L. Van Gool. Color-based object tracking in multi-camera environment. In *25th Pattern Recognition Symposium, DAGM*, Magdeburg, Germany, 2003.

[15] M. Paskin, C. Guestrin, and J. McFadden. A robust architecture for distributed inference in sensor networks. In *the Fourth International Conference on Information Processing in Sensor Networks (IPSN'05)*, 2005.

[16] E. Sudderth, A. Ihler, W. Freeman, and A. Willsky. Nonparametric belief propagation. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 605–612, Madison, WI, 2003.

[17] J. Sun, N. Zheng, and S. Harry. Stereo matching using belief propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(7):787–800, 2003.

[18] J. Vermaak, N. Lawrence, and P. Pérez. Variational inference for visual tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 605–612, Madison, WI, 2003.

[19] Y. Wang, J. Wu, and A. Kassim. Multiple cameras tracking using particle filtering. In *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, pages 33–39, Breckenridge, CO, 2005.

[20] J. Winn and C. Bishop. Variational message passing. *Journal of Machine Learning Research*, (5), 2004.

[21] J. Yedidia, W. Freeman, and Y. Weiss. Generalized belief propagation. *Advances in Neural Information Processing Systems (NIPS)*, 13:689–695, 2000.