

The following paper posted here is not the official IEEE  
published version.

The final published version of this paper can be found in the  
Proceedings of TENCON (2006 : Hong Kong, China):pp.1-4

Copyright © 2006 IEEE.

Personal use of this material is permitted. However, permission to  
reprint/republish this material for advertising or promotional purposes or  
for creating new collective works for resale or redistribution to servers or  
lists, or to reuse any copyrighted component of this work in other works  
must be obtained from the IEEE.

# Distributive JPDAF for Multi-Target Tracking in Wireless Sensor Networks

Hui Ma, Brian W.-H. Ng  
The University of Adelaide  
SA 5005, Australia

**Abstract-** In this paper we present the development of a distributive joint probabilistic data association filter (JPDAF) for multi-target tracking in wireless sensor networks. We adopt sequential Monte Carlo (SMC) method to implement the JPDAF, and use Gaussian mixture model (GMM) to develop the distributive JPDAF. Simulation results are also provided.

## I. INTRODUCTION

Multi-target tracking is one of the representative applications of wireless sensor networks (Figure 1): a large number of sensor nodes collaboratively sense and infer multiple targets' states (*e.g.*, position, velocity and heading). For the centralised systems (*e.g.* radar, sonar...etc.), the multi-target tracking techniques are well-established [1], [6]. However, for wireless sensor networks, their unique characteristics pose significant challenges in developing target tracking algorithms: the tracking algorithms need to consider the interplay between data and information processing and sensor network architecture; and the tracking algorithms need to be distributive and consume less computation and communication resources [1], [3], [5].

In this paper, the sequential Monte Carlo (SMC) method, also named as Particle filtering [2] is adopted for the algorithmic development. Several SMC based tracking algorithms for wireless sensor networks have been proposed in literature [3]-[5]. However, these algorithms are normally applied to track single target or track multiple targets that are sufficiently separated in space and/or time or track multiple targets using classification techniques. They might not be readily applied to the tracking scenario as depicted in Figure 1, in which two targets traverse closely. To address this problem, we have developed distributive joint probabilistic data association filter (JPDAF). And SMC is adopted for the implementation of this JPDAF.

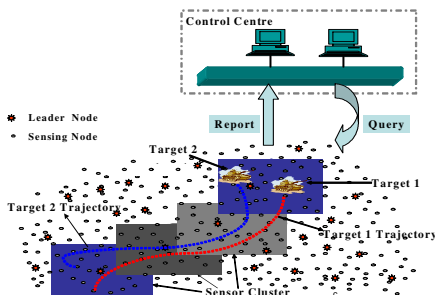


Figure 1. Tracking two close-spaced targets in a wireless sensor network

The remainder of this paper is organized as follows. In Section II, we briefly review the SMC method. In Section

III, we formulate the multi-target tracking problem in wireless sensor networks. In Section IV, we detail the design of SMC implemented JPDAF. In section V, we develop the distributive SMC-JPDAF by adopting Gaussian mixture model (GMM). We show the simulation results in Section VI and conclude the paper in section VII.

## II. SEQUENTIAL MONTE CARLO (SMC) METHOD

In general, Bayesian estimation recursively calculates the posterior probability distribution of the state vector  $x_t$  given the measurement vectors up to time  $t$  (denoted as  $z_{0:t} = \{z_0, z_1, \dots, z_t\}$ ) in two steps:

*Prediction step:*

$$p(x_t | z_{0:t-1}) = \int p(x_t | x_{t-1}) p(x_{t-1} | z_{0:t-1}) dx_{t-1} \quad (1)$$

*Filtering step:*

$$p(x_t | z_{0:t}) \propto p(z_t | x_t) p(x_t | z_{0:t-1}) \quad (2)$$

The SMC method represents the posterior distribution  $p(x_t | z_{0:t})$  as a set of weighted random samples, called particles  $\{x_t^i, \omega_t^i\}_{i=1}^N$  [2]:

$$p(x_t | z_{0:t}) \approx \sum_{i=1}^{N_t} \omega_t^i \delta(x_t - x_t^i) \quad (3)$$

where  $\delta(\cdot)$  is the Dirac delta function. The weights of each particle are updated sequentially by:

$$\omega_t^i = \omega_{t-1}^i \frac{p(z_t | x_t^i) p(x_t^i | x_{t-1}^i)}{\pi(x_t^i | x_{t-1}^i, z_{0:t})} \quad (4)$$

where  $\pi(x_t | x_{t-1}, z_{0:t})$  is called the *proposal distribution* that is known and easier to sample [2]. As the number of samples becomes large, these samples effectively provide an equivalent representation of the posterior distribution. SMC method could be used in general nonlinear, non-Gaussian systems [2].

## III. THE FORMULATION OF MULTI-TARGET TRACKING IN WIRELESS SENSOR NETWORKS

In this paper, we assume the number of targets  $K$  is known and fixed. We also make assumption that there are

two types of sensor nodes (Figure 1): the *leader nodes* which are responsible for information processing; and the *sensing nodes* which provide leader nodes with measurements regarding targets states. At time step  $t$ , there are  $N_t$  sensing nodes which participate in the tracking task.

#### A. System Model and Measurement Model

In multi-target tracking, it is generally assumed that each individual target evolves independently [6]. Thus, at time step  $t$ , the joint state vector  $x_t = (x_t^1, \dots, x_t^K)$  can be decomposed into  $K$  partial equations:

$$x_t^k = f_t^k(x_{t-1}^k, v_t^k) \quad \forall k = 1, \dots, K \quad (5)$$

The noises  $v_t^k$  is assumed to be Gaussian and independent of different targets.

The tracking algorithms developed in this paper are based on acoustic energy measurement at individual sensing nodes. When the  $k$ th target passes the  $n$ th sensing node, the measurement obtained at  $n$ th node is [5]:

$$z_k^n(t) = \frac{S_k(t)}{\|\xi_k(t) - \rho^n\|^2} + \varepsilon^n(t) \quad (6)$$

where  $S_k(t)$  is the acoustic intensity generated by the  $k$ th target, and  $\varepsilon^n \sim N(\mu^n, \sigma^n)$  is the additive Gaussian noise.  $\xi_k(t)$  and  $\rho^n$  are the position coordinates of the  $k$ th target and the  $n$ th sensing node, respectively.

In the presence of multiple targets, the measurements obtained at the  $n$ th sensing node are designated by a vector  $z^n = (z_1^n, \dots, z_{M^n}^n)$ . The total number of measurements  $M^n$  comprises the measurements that arise from targets  $M_T^n$ , and the measurements that are due to the clutter (false alarm)  $M_C^n$ . These measurements are assumed to be independent of each other and independent of those at the other sensing nodes. Here, we also make the assumptions that, one target can generate zero or one measurement at one time; and one measurement can originate from one target or from the clutter.

#### B. Data Association

In multi-target tracking, the measurements are normally unlabelled and we do not know which measurement is generated by which target (or clutter). Therefore, we need to define the target to measurement association hypothesis:  $\tilde{\lambda} = (\tilde{\lambda}^1 \dots \tilde{\lambda}^{N_t})$ , where  $\tilde{\lambda}^n = (\tilde{r}^n, M_C^n, M_T^n)$  is for  $n$ th sensing node at time step  $t$ .  $\tilde{r}^n$  is the target to measurement association vector and its component  $\tilde{r}_k^n$  (for  $k$ th target) is a random variable defined as below:

$$\tilde{r}_k^n = \begin{cases} 0 & \text{if target } k \text{ is undetected} \\ j \in \{1 \dots M_T^n\} & \text{if target } k \text{ generates measurement } j \end{cases} \quad (7)$$

## IV. THE SMC IMPLEMENTATION OF JPDAF

The JPDAF is one of the popular strategies in target tracking research community [6], [8]. In [7], the authors used SMC method to implement JPDAF. In this paper we adopt a similar approach and extend it to distributive multi-target tracking in wireless sensor networks. In this section, we briefly describe both the JPDAF and its SMC implementation, the SMC-JPDAF. In the next section, we present distributive SMC-JPDAF.

#### A. General JPDAF Strategy

For each target, the prediction step in JPDAF is preceded independently as below:

$$p_k(x_{k,t} | z_{0:t-1}) = \int p_k(x_{k,t} | x_{k,t-1}) p(x_{k,t-1} | z_{0:t-1}) dx_{k,t-1} \quad (8)$$

The filtering step is:

$$p_k(x_{k,t} | z_{0:t}) \propto p_k(z_t | x_{k,t}) p_k(x_{k,t} | z_{0:t-1}) \quad (9)$$

However, the filtering step cannot be performed independently for each target. This is because the likelihood of  $k$ th target  $p_k(z_t | x_{k,t})$  could not be calculated independently for each target due to the data association. In JPDAF,  $p_k(z_t | x_{k,t})$  is calculated as follows [6], [7]:

$$p_k(z_t | x_{k,t}) = \prod_n \left[ \beta_{0,k}^n + \sum_j \beta_{j,k}^n p_T^n(z_{j,t}^n | x_{k,t}) \right] \quad (10)$$

where  $p_T^n(z_{j,t}^n | x_{k,t})$  is the likelihood of  $j$ th measurement with respect to  $k$ th target at sensing node  $n$ .  $\beta_{j,k}^n$  is the marginal association probability of the  $k$ th target with respect to the  $j$ th measurement.  $\beta_{0,k}^n$  is the undetected probability of the  $k$ th target.  $\beta_{j,k}^n$  (for the measurements from both targets and clutter) can be acquired by summing over all valid joint association probabilities  $\eta_i^n$  as:

$$\beta_{j,k}^n = p(\tilde{r}_{k,t}^n = j | z_{0:t}) = \sum_{\tilde{\lambda}_t^n \in \eta_t^n} p(\tilde{\lambda}_t^n | z_{0:t}) \quad j=0, \dots, M^n \quad (11)$$

where the individual joint association probability  $p(\tilde{\lambda}_t^n | z_{0:t})$  could be further calculated as:

$$p(\tilde{\lambda}_t^n | z_{0:t}) = P_D^{K-\sigma} (1-P_D)^\sigma P_F^{M^n-(K-\sigma)} \prod_{j \in \{1, 2, \dots, K-\sigma\}} p_k(z_{j,t}^n | z_{0:t-1}) \quad (12)$$

where  $P_D$  denotes the probability of detection,  $P_F$  denotes the probability of false alarm.  $\sigma$  is the number of false alarm.  $p_k(z_{j,t}^n | z_{0:t-1})$  is the predictive likelihood for the

$j$  th measurement using the information from the  $k$  th target at the  $n$  th node and is expressed as:

$$p_k(z_{j,t}^n | z_{0:t-1}) = \int p_T^n(z_{j,t}^n | x_{k,t}) p(x_{k,t} | z_{0:t-1}) dx_{k,t} \quad (13)$$

We assume that the predictive likelihood is independent over the sensing nodes, as well as over the individual measurements at each sensing nodes.

For more details of the above association probabilities calculations, readers may refer to [8].

### B. The SMC Implementation of JPDAF

For the  $k$  - th target, we assume that there is a set of samples  $\{x_{k,t-1}^i, \omega_{k,t-1}^i\}_{i=1}^{N_s}$  approximating the posterior distribution  $p_k(x_{k,t-1} | z_{1:t-1})$  at the previous time step  $t-1$ . At the current time step  $t$ , the new samples for the target state are generated from *transition prior*:

$$x_{k,t}^i \sim p_k(x_{k,t}^i | x_{k,t-1}^i) \quad (14)$$

Then the predictive likelihood in (13) can be straightforwardly approximated [2], [7], [10]:

$$p_k(z_{j,t}^n | z_{0:t-1}) \approx \sum_{i=1}^{N_s} p_T^i(z_{j,t}^n | x_{k,t}^i) \quad (15)$$

$p_T^i(z_{j,t}^n | x_{k,t}^i)$  in both Equation (10) and (15) could be computed according to Equation (6).

By substituting Equation (15) into (12), we can get the approximations for the valid joint association probabilities. Then these joint association probabilities could be fed into Equation (11) to compute the marginal association probabilities. And in turn, we can approximate the target likelihood by using Equation (10). Finally, this target likelihood is used to set the new weights of particles:

$$\omega_{k,t}^i \propto \omega_{k,t-1}^i p_k(z_t | x_{k,t}^i) \quad (16)$$

So finally we get a new sample set  $\{x_{k,t}^i, \omega_{k,t}^i\}_{i=1}^{N_s}$  that approximates the posterior distribution  $p_k(x_{k,t} | z_{0:t})$  at current time step. The SMC implementation of JPDAF is summarized in Algorithm 1.

### V. DISTRIBUTIVE SMC IMPLEMENTED JPDAF

We partition the whole sensor field into smaller regions and in each smaller region, a group of sensor nodes form a sensor cluster [10]. When a leader node detects the approaching targets, it collects measurements from its neighboring sensing nodes and runs JPDAF to estimate the target's states. When the target moves out of current sensor node cluster, the leader node propagates its last estimation results, e.g. the posterior distribution  $p(x_k | z_{0:k})$  to the next leader node.

However, rather than propagating particles or weights (up to 1000 particles per target in the simulation), the posterior distribution can be propagated through the transmission of lower dimensional Gaussian mixture model (GMM):

$$p(x_k | z_{0:k}) \approx \hat{p}(x_k | z_{0:k}) = \sum_{g=1}^G \alpha^g N(\mu^g, \mathbf{P}^g) \quad (17)$$

where  $G$  is the number of mixing components, and  $\alpha^g$  are the mixing weights and  $N(\mu^g, \mathbf{P}^g)$  are Gaussian density. With GMM approximation, for each target we only need to transmit 15 parameters. The distributive SMC-JPDAF is summarized in Algorithm 2.

#### Algorithm 1 SMC-JPDAF

- A. At  $t=0$ , for  $k=1, 2, \dots, K$  and  $i=1, \dots, N_s$  draw particles  $x_{k,0}^i \sim p_k(x_{k,0})$ .
- B. For  $t=1, \dots, T$ , do the following:
  1. Sample  $x_{k,t}^i \sim p(x_{k,t}^i | x_{k,t-1}^i)$
  2. For  $k=1, 2, \dots, K, i=1, \dots, N_s, j=1, 2, \dots, M^n$ , compute the predictive weights – Equation (15).
  3. For  $i=1, 2, \dots, N_s$ , enumerate all the valid joint target to measurement association hypotheses at the  $n$  th sensing node.
  4. For  $i=1, 2, \dots, N_s$ , compute all the joint association posterior probability – Equation (12).
  5. For  $k=1, 2, \dots, K, i=1, 2, \dots, N_s, j=1, 2, \dots, M^n$ , compute the marginal association probability – Equation (11).
  6. For  $k=1, 2, \dots, K$  and  $i=1, \dots, N_s$  compute the target likelihood – Equation (10).
  7. For  $k=1, 2, \dots, K$  and  $i=1, \dots, N_s$  compute and normalize the particle weights – Equation (16).

#### Algorithm 2 Distributive SMC-JPDAF

1. The initial leader node  $a$  does the following:
  - 1.1 for  $k=1, 2, \dots, K$  and  $i=1, \dots, N_s$  draw particle  $x_{k,0}^i \sim p_k(x_{k,0})$ .
  - 1.2. Select sensing nodes and get their measurements.
  - 1.3. Calculate the posterior  $p(x_k | z_{0:k})$  by Algorithm 1.
  - 1.4. Check if the targets have moved to the next region. If “yes”, do 1.5 and 1.6; else repeat 1.2. and 1.3.
  - 1.5. Compute the GMM of posterior  $p(x_k | z_{0:k})$ .
  - 1.6. Forward GMM parameters  $(\alpha^g, \mu^g, \mathbf{P}^g)$  to leader node  $b$ .
2. The leader node  $b$  does the following:
  - 2.1. Redraw posterior PDF by sampling the GMM  $(\alpha^g, \mu^g, \mathbf{P}^g)$ .
  - 2.2. ~ 2.5. Same as 1.2. ~ 1.5
  - 2.6. Forward GMM parameters  $(\alpha^g, \mu^g, \mathbf{P}^g)$  to leader node  $c$ .
3. The leader node  $c$  repeats the steps as above.
  - ...

## VI. SIMULATION

Figure 2 shows the simulation set-up, including the sensor field, ground truth (two vehicles travel from north-east to south-west), and the locations of 20 sensor nodes which are organized in two cliques (clusters).

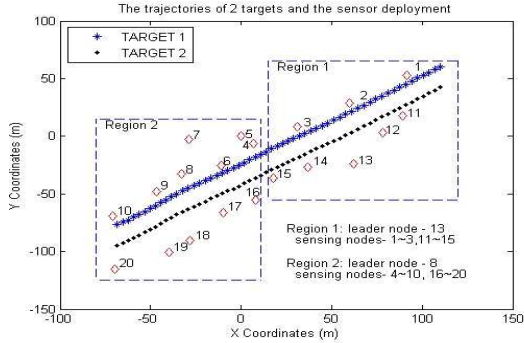


Figure 2. Simulation set-up for two targets tracking in a WSN

The target state vector at time  $t$  is  $x_t$  which is composed of includes the target positions  $x$  and  $y$ , and the target velocities  $v_x$  and  $v_y$ . We use a nearly constant velocity model to model the dynamics of the targets [6], [10]. At each time step, the acoustic energy measurements at the sensor node  $n$  are obtained by Equation (6) in which the measurement noise  $\varepsilon_n$  is Gaussian with zero mean and variance  $R = 0.5$ , and  $S$  is set to 5000. We use 1000 particles per target in simulation. In the simulation, we assume there is no clutter and the detection rate is set to 95% for all sensing nodes.

We repeated 50 runs for the developed algorithms with the same ground truth. The root mean square error (RMSE) is used to compare the tracking accuracy of the different algorithm. Figure 3 shows the tracking results that are averaged over 50 runs. Figure 4 shows the RMSE values of 50 runs for target 1 and target 2 respectively. From Figures 3 and 4, it can be seen that the distributive SMC-JPDAF tracking algorithm can attain very promising tracking accuracy. It also significantly conserves communication bandwidth by adopting GMM approximation. Furthermore, the computation cost of the distributive SMC-JPDAF is moderate.

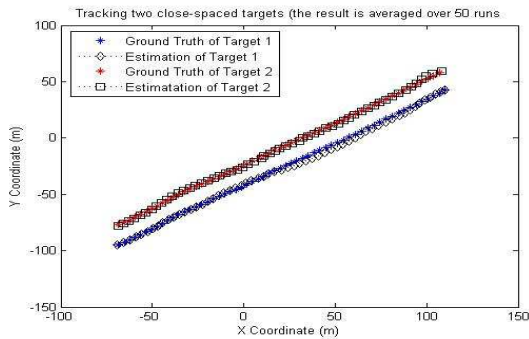


Figure 3. Estimation results of two targets

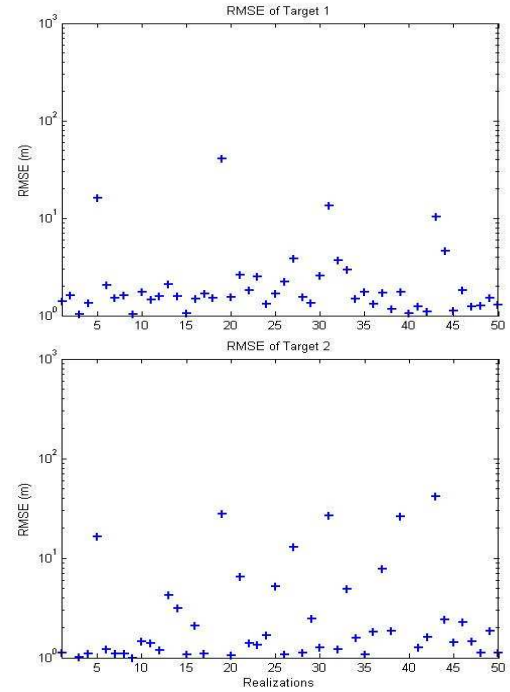


Figure 4 RMSE of target 1 and target 2

## VII. CONCLUSION

In this paper, we report the development of the distributive SMC implemented JPDAF for multi-target tracking in wireless sensor networks. The simulation results show that the developed algorithm could attain accurate tracking accuracy at lower communication cost.

### References

- [1] C.V.Chong and S.P.Kumar, "Sensor Networks: Evolution, Opportunities, and Challenges", in Proceedings of the IEEE, Vol. 91, No.8, 2003.
- [2] A.Doucet, J.deFreitas, and N. Gordon, Eds., "Sequential Monte-Carlo Methods in Practice," Springer-Verlag, 2001.
- [3] F.Zhao and J.Shin and J.Reich, "Information-Driven Dynamic Sensor Collaboration for Tracking Applications", Proceedings of the IEEE, Vol. 91, No. 8, 2003.
- [4] Mark Coates, "Distributed particle filters for sensor networks", IPSN 2003, Springer, 2004.
- [5] Xiaohong Sheng, Yu-Hen Hu and Parameswqran Ramanathan, "Distributed Partilce Filter with GMM Approximation for Mulitple Targets Localization and Tracking in Wireless Sensor Network", IPSN 2005.
- [6] Y.Bar-Shalom abd T.E.Fortmann, "Tracking and Data Association," Academic Press, 1988.
- [7] Jaco Vermaak, Simon J. Godsill and Partrick Perez, "Monte Carlo Filtering and Multi-Target Tracking and Data Association, " IEEE Transactions on Aerespace and electronic systems, VOL. 41, NO. 1, 2005.
- [8] Samuel Blackman and Robert Popoli, "Design and Analysis of Modern Tracking Systems," Artech House, 1999.
- [9] Oliver Frank, "Multiple Target Tracking", thesis, Swiss Federal Institute of Technology, February 2003.
- [10] H.Ma and Brian W.-H.Ng, "Collaborative Data and Information Processing for Target Tracking in Wireless Sensor Networks," 4<sup>th</sup> IEEE Conference on Industrial Informatics, 15-18 August 2006, Singapore.