Multi-Target Tracking in Wireless Sensor Networks using Distributed Joint Probabilistic Data Association and Average Consensus Filter

M. A. Tinati, T. Yousefi Rezaii
Electrical and Computer Engineering Department
Tabriz University, Tabriz, Iran
{tinati and yousefi}@tabrizu.ac.ir

Abstract

The aim of this paper is to develop a distributed Multi-Target Tracking (MTT) algorithm over wireless sensor networks which has the ability of online implementation in low-cost sensor nodes due to its lower computational complexity and execution time compared to the other MTT systems. The Monte Carlo (MC) implementation of JPDAF (MC-JPDAF) is applied to the classical problem of MTT in a cluttered area. Also, to make the tracking algorithm scalable and usable for large networks, the distributed Expectation Maximization (EM) algorithm is used via the average consensus filter in order to diffuse the nodes' information over the whole network. Furthermore, some simplifications and modifications are made to MC-JPDAF algorithm in order to reduce the computation complexity of the tracking system and make it suitable for low-energy sensor networks. Finally, the simulations of tracking tasks for the proposed system are given.

1. Introduction

Current technology has enabled the development of distributed ad-hoc networks of hundreds or thousands of nodes, each capable of sensing, processing and communication. The ability to track targets is essential in many applications, such as military applications including missile defense and battlefield situational awareness. So far, much of the theory of tracking is developed for centralized processing of data from a relatively small number of radars or similar large devices endowed with plenty of power and high-bandwidth communications. In this paper we develop the MTT problems in distributed manner in order to simultaneously make the tracking system scalable and usable for large networks while providing better tracking performance by using the spatial diversity obtained by spreading the sensors in the desired area to be covered. The MTT problem is not a trivial extension of single target tracking but rather a challenging topic of research. In MTT scenarios, there is a combinatorial explosion in the space of possible multiple target trajectories due to the uncertainty in the association of observed measurements with known targets in each step and the main challenge in tracking multiple targets is to manage the computational complexity of the problem while still providing reasonable tracking performance.

Recently, the JPDAF approach to solve the data association problem has gained more popularity due to its low computational complexity and online implementation. But the main shortcoming of the JPDAF is that, to maintain the tractability, the final estimate collapses to a single Gaussian, thus discarding pertinent information. The linear and Gaussian models assumptions are often made by some authors to simplify hypotheses evaluation for target originated measurements. The implementation of JPDAF using the Extended Kalman Filter (EKF) is an instance [1]. However, the performance of these algorithms degrades as the non-linearities become more severe. More recently, strategies have been proposed to combine the JPDAF with Particle techniques to accommodate general nonlinear and non-Gaussian models [2-3]. The MC-JPDAF developed in [3], can be considered as the first comprehensive algorithm that uses the Monte Carlo methods to implement the MTT while efficiently taking into account the data association problem.

In order to design a distributed MTT algorithm, it is necessary to map an MTT solution onto a sensor network platform with diverse resource limitations, including power, sensing, communication and computation. Currently, several distributed target tracking algorithms have been developed [4-5]. In [4], the distributed particle filter is based on factorizing the likelihood and forming parametric approximations to products of likelihood factors. The model parameters

978-0-7695-3516-6/08 $25.00 © 2008 IEEE
DOI 10.1109/ICACC.2009.62

are then exchanged between sensor nodes instead of the data or exact particle information. Using a Gaussian Mixture Model to approximate the posterior distribution is proposed in [5], where the estimated parameters of GMM are transmitted to a fusion center. In [6] the distributed EM algorithm is developed and used to approximate the global GMM parameters for the posterior distribution of weighted particles. However, this method is only designed for single target tracking situations, where the algorithm doesn't encounter with data association problem which is common in multi-target scenarios.

In this paper we have developed a distributed multi-target tracking algorithm using the simplified version of MC-JPDAF, in order to solve the problem of MTT for the general case of nonlinear and non-Gaussian models, which we call it SMC-JPDAF. In the proposed method, each sensor estimates the marginal posterior densities of the individual targets through the Monte Carlo simulations and then calculates the local GMM parameters for these distributions. In continue each node obtains the global GMM parameters for the considered distributions by using the average consensus filtering technique. This iterative procedure is repeated for consecutive time steps, where at each time step, the global approximations for the individual posterior densities are obtained from each node.

2. SMC-JPDAF framework

As mentioned previously, the JPDAF approach is the most widely applied method in MTT problems considering the data association uncertainty due to its simplicity and lower computational complexity in contrast with the other approaches. For this reason we have chosen this approach to solve the data association problem in this paper. Since the target dynamics and measurement likelihood models in target tracking applications are nonlinear and non-Gaussian in general, the selected Bayesian framework should have the ability to estimate and track such models. For this reason, the simplified version of the Monte Carlo implementation of JPDAF is used in our distributed MTT system. Due to the Particle filtering methods used in MC-JPDAF, it has the ability to track the arbitrary proposal distributions. In the original MC-JPDAF given in [3], a relatively complicated algorithm is developed to obtain the optimum proposal density for each target at each observer sensor, which is only reasonable for centralized tracking systems as in [3]. In this paper we propose to use the prior distribution rather that the distribution given in [3] in order to significantly reduce the computational cost and make the tracking algorithm suitable for distributive and online implementation. The reductions in computational costs appear as the simplification of the update equations obtained for predictive coefficients and the importance weights and also the proposal density. The main idea of JPDAF is to recursively update the marginal filtering distributions for each of the targets $p_k(\mathbf{x}_{k,t} \mid \mathbf{z}_{1,t-1})$, $k = 1,...,K$, through the Bayesian sequential estimation recursion as:

$$p_k(\mathbf{x}_{k,t} \mid \mathbf{z}_{1,t-1}) = \int p_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}) p_k(\mathbf{x}_{k,t-1} \mid \mathbf{z}_{1,t-1}) d\mathbf{x}_{k,t-1} \tag{1}$$

where $\mathbf{x}_{k,t}$ is the state of the k^{th} target at time *t*, $\mathbf{z}_{1,t-1}$ are the obtained observation data until time *t-1*, $p_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1})$ is the prior density of the k^{th} target and $p_k(\mathbf{x}_{k,t-1} \mid \mathbf{z}_{1,t-1})$ is the likelihood of the k^{th} target. Due to the data association uncertainty, the filtering step cannot be performed independently. In JPDAF the likelihood for the k^{th} target is assumed to be:

$$p_k(\mathbf{z}_t \mid \mathbf{x}_{k,t}) = \prod_{i=1}^{N_o} \left[\beta_{jk}^i + \sum_{j=1}^{M^i} \beta_{jk}^i p_T^i(\mathbf{z}_{j,t}^i \mid \mathbf{x}_{k,t}) \right] \tag{2}$$

where $\beta_{jk}^i = p(\tilde{r}_{k,t}^i = j \mid \mathbf{z}_{1:t}),\ j = 1...M^i$, is the posterior probability that indicates how the k^{th} target is associated with the j^{th} measurement in the i^{th} observer, and β_{0k}^i is the posterior probability that the k^{th} target is undetected. $p_T^i(\mathbf{z}_{j,t}^i \mid \mathbf{x}_{k,t})$ is the likelihood of the kth target under the hypothesis that it is associated with the ith observer. Furthermore, the likelihood is assumed to be independent over the observers. With the definition of the likelihood as in (2), the filtering step is:

$$p_k(\mathbf{x}_{k,t} \mid \mathbf{z}_{1:t}) \propto p_k(\mathbf{z}_t \mid \mathbf{x}_{k,t}) p_k(\mathbf{x}_{k,t} \mid \mathbf{z}_{1:t-1}) \tag{3}$$

All that remains is to compute the posterior probabilities of the marginal associations β_{jk}^i as:

$$\beta_{jk}^i = p(\tilde{r}_{k,t}^i = j \mid \mathbf{z}_{1:t}) = \sum_{\{\tilde{\lambda}_t^i \in \tilde{\Lambda}_t^i : \tilde{r}_{k,t}^i = j\}} p(\tilde{\lambda}_t^i \mid \mathbf{z}_{1:t}) \tag{4}$$

where $\tilde{\Lambda}_t^i$ is the set of all joint target to measurement association hypotheses for the data at the i^{th} observer. The posterior probability for the joint association hypothesis $p(\tilde{\lambda}_t^i \mid \mathbf{z}_{1:t})$ can be expressed as [3]:

$$p(\tilde{\lambda}_t^i \mid \mathbf{z}_{1:t}) \propto p(\tilde{\lambda}_t^i)(V^i)^{-M_C^i} \prod_{j \in \mathrm{I}^i} p_{r_{j,t}^i}(\mathbf{z}_{j,t}^i \mid \mathbf{z}_{1:t-1}) \tag{5}$$

where the expression $p(\tilde{\lambda}_t^i)$ is given in [3], V^i is the volume of the measurement space for the i^{th} observer defined as $V^i = 2\pi R_{\max}^i$, where $R_{\max}^i$ is the maximum range for the i^{th} observer, and $\mathrm{I}^i = \{j \in \{1...M^i\} : r_j^i \neq 0\}$. The expression

$p_{r^i_{j,t}}(\mathbf{z}^i_{j,t} \mid \mathbf{z}_{1:t-1})$ in (5) is the predictive likelihood for the j^{th} measurement at the i^{th} observer using the information from the k^{th} target, given in the standard form by:

$$p_k(\mathbf{z}^i_{j,t} \mid \mathbf{z}_{1:t-1}) = \int p^i_T(\mathbf{z}^i_{j,t} \mid \mathbf{x}_{k,t}) p_k(\mathbf{x}_{k,t} \mid \mathbf{z}_{1:t-1}) d\mathbf{x}_{k,t} \tag{6}$$

In Monte Carlo frame work, the predictive likelihood in (5) is approximated using the Monte Carlo samples from the proposal distribution. It is assumed that for the k^{th} target the set of samples $\{w^{(n)}_{k,t-1}, \mathbf{x}^{(n)}_{k,t-1}\}^N_{n=1}$ are available, approximately distributed according the marginal filtering distribution at the previous time step, $p_k(\mathbf{x}_{k,t-1} \mid \mathbf{z}_{1:t-1})$. At the current time step, the new samples for the target state are generated from a suitably constructed proposal distribution, i.e.:

$$\mathbf{x}^{(n)}_{k,t} \sim q_k(\mathbf{x}_{k,t} \mid \mathbf{x}^{(n)}_{k,t-1}, \mathbf{z}_t), \quad n = 1 \ldots N \tag{7}$$

In our distributed MTT system, in order to reduce the computational complexity of the proposed tracking algorithm, we have used the prior distribution as the proposal distribution for k^{th} target rather than the proposed one in [3]. This will significantly reduce the execution time of the tracking algorithm, as discussed in the simulations later. So the proposal distribution will be of the form:

$$q_k(\mathbf{x}_{k,t} \mid \mathbf{x}^{(n)}_{k,t-1}, \mathbf{z}_t) = p_k(\mathbf{x}^{(n)}_{k,t} \mid \mathbf{x}^{(n)}_{k,t-1}) \tag{8}$$

Using these Monte Carlo samples, the predictive likelihood in (5) can be approximated as:

$$p_k(\mathbf{z}^i_{j,t} \mid \mathbf{z}_{1:t-1}) \approx \sum_{n=1}^{N} \alpha^{(n)}_{k,t} p^i_T(\mathbf{z}^i_{j,t} \mid \mathbf{x}^{(n)}_{k,t}) \tag{9}$$

where, using the definition in (8), the predictive weights are equal to importance weights, $\alpha^{(n)}_{k,t} = w^{(n)}_{k,t}$, $n = 1, \ldots, N$. So the update equation for the importance weights is as:

$$\alpha^{(n)}_{k,t} \propto w^{(n)}_{k,t-1} \frac{p_k(\mathbf{x}^{(n)}_{k,t} \mid \mathbf{x}^{(n)}_{k,t-1})}{q_k(\mathbf{x}^{(n)}_{k,t} \mid \mathbf{x}^{(n)}_{k,t-1}, \mathbf{z}_t)}, \quad \sum_{n=1}^{N} \alpha^{(n)}_{k,t} = 1 \tag{10}$$

The approximation to the predictive likelihood can now be substituted into (5) to obtain the approximation for the joint association posterior probabilities, from which approximations for the marginal target to measurement association posterior probabilities can be computed according to (4). These approximations can be used in (3) to approximate the target likelihood. Finally, setting the new importance weights to:

$$w^{(n)}_{k,t} \propto w^{(n)}_{k,t-1} p_k(\mathbf{z}_t \mid \mathbf{x}^{(n)}_{k,t}), \quad \sum_{n=1}^{N} w^{(n)}_{k,t} = 1 \tag{11}$$

leads to the sample set $\{w^{(n)}_{k,t}, \mathbf{x}^{(n)}_{k,t}\}^N_{n=1}$ being approximately distributed according to the marginal filtering distribution at the current time step $p_k(\mathbf{x}_{k,t} \mid \mathbf{z}_{1:t})$.

3. Formulation of distributed EM algorithm for Gaussian mixtures of multiple targets used in proposed MTT system

In this section, we will not be using the time index t anymore for simplicity. The obtained equations can be used for the consecutive time steps. The Expectation Maximization algorithm for Gaussian mixtures in distributed manner deals with calculation of the local sufficient statistics in all observers for each target and then using these local sufficient statistics and the neighbor sensors global sufficient statistics of the previous time to estimate the global summary statistic for the next time step in each observer. For this reason, a sensor network with K targets to be tracked, J independent observers, M^j measurements at the j^{th} observer and N particles as the number of samples to be generated for each target in each time step, are assumed. The desired posterior distribution for each target is assumed to be a Gaussian mixture with C mixture probabilities $\alpha_{k,j,c}$. The Gaussian mixture distribution for state $\mathbf{x}^{(n)}_{k,j}$ is:

$$p(\mathbf{x}^{(n)}_{k,j} \mid \boldsymbol{\psi}_k) = \sum_{c=1}^{C} \alpha_{k,j,c} p(\mathbf{x}^{(n)}_{k,j} \mid \boldsymbol{\mu}_{k,c} \boldsymbol{\Sigma}_{k,c}) \tag{12}$$

where $\boldsymbol{\psi}_k$ is the set of the distribution parameters to be estimated for target k, where $\boldsymbol{\psi}_k = \{\alpha_{k,j,c}, \boldsymbol{\mu}_{k,c}, \boldsymbol{\Sigma}_{k,c}\}$.

The probability for the state $\mathbf{x}^{(n)}_{k,j}$ in c^{th} mixture follows a Gaussian distribution with mean $\boldsymbol{\mu}_{k,c}$ and variance $\boldsymbol{\Sigma}_{k,c}$:

$$p(\mathbf{x}^{(n)}_{k,j} \mid \boldsymbol{\mu}_{k,c}, \boldsymbol{\Sigma}_{k,c}) = \frac{1}{\sqrt{2\pi} \left|\boldsymbol{\Sigma}_{k,c}\right|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}^{(n)}_{k,j} - \boldsymbol{\mu}_{k,c})^T \boldsymbol{\Sigma}^{-1}_{k,c} (\mathbf{x}^{(n)}_{k,j} - \boldsymbol{\mu}_{k,c})} \tag{13}$$

The local summary quantities for the k^{th} target at the j^{th} observer will become:

$$\alpha^q_{k,j,c} = \frac{1}{N} \sum_{n=1}^{N} \alpha^q_{k,j,c,n} \tag{14}$$

$$\boldsymbol{\beta}_{k,j,c}^{q} = \frac{1}{N}\sum_{n=1}^{N}\alpha_{k,j,c,n}^{q}\mathbf{x}_{k,j}^{(n)} \tag{15}$$

$$\boldsymbol{\gamma}_{k,j,c}^{q} = \frac{1}{N}\sum_{n=1}^{N}\alpha_{k,j,c,n}^{q}(\mathbf{x}_{k,j}^{(n)} - \boldsymbol{\mu}_{k,j,c}^{q})(\mathbf{x}_{k,j}^{(n)} - \boldsymbol{\mu}_{k,j,c}^{q})^{T} \tag{16}$$

where the index q in (14) to (16) denotes the iteration number for the distributed GMM algorithm and $\boldsymbol{\mu}_{k,j,c}^{q}$ is the local mean in the j^{th} observer for the c^{th} mixture component. The expression for the posterior probability $\alpha_{k,j,c,n}^{q}$ is obtained from the set of distribution parameters in the previous time step $\boldsymbol{\psi}_{k}^{q-1}$ as follows:

$$\alpha_{k,j,c,n}^{q} = \frac{\alpha_{k,j,c}^{q-1}p(\mathbf{x}_{k,j}^{(n)} \mid \boldsymbol{\mu}_{k,c}^{q-1}, \boldsymbol{\Sigma}_{k,c}^{q-1})}{\sum_{l=1}^{C}\alpha_{k,j,l}^{q-1}p(\mathbf{x}_{k,j}^{(n)} \mid \boldsymbol{\mu}_{k,l}^{q-1}, \boldsymbol{\Sigma}_{k,l}^{q-1})} \tag{17}$$

The calculation stage of the local summary quantities is called expectation step (E-step). In continue the global summary quantities for each sensor can be obtained using the local summary quantities of that sensor and the global summary quantities of its neighboring sensors. We use $\mathbf{u}_{k,j,c}^{q} = [N\alpha_{k,j,c}^{q}, \boldsymbol{\beta}_{k,j,c}^{q}, \boldsymbol{\gamma}_{k,j,c}^{q}]^{T}$ to denote the local summary quantities in node j. Also we use $\boldsymbol{\chi}_{k,j,c}^{q} = [\boldsymbol{\chi}_{k,j,c}^{q}(1), \boldsymbol{\chi}_{k,j,c}^{q}(2), \boldsymbol{\chi}_{k,j,c}^{q}(3)]^{T}$ to denote the global summary quantities in node j. Now, the discrete form of the consensus filter can be used to obtain approximations for the global summary quantities in each node (Consensus step) as follows:

$$\boldsymbol{\chi}_{k,j,c}^{q+1} = \boldsymbol{\chi}_{k,j,c}^{q} + \delta_{j}\left[\sum_{m\in N_{j}}(\boldsymbol{\chi}_{k,m,c}^{q} - \boldsymbol{\chi}_{k,j,c}^{q}) + (\mathbf{u}_{k,j,c}^{q} - \boldsymbol{\chi}_{k,j,c}^{q})\right] \tag{18}$$

where δ_{j} is the step size parameter for the j^{th} observer, determining the updating rate and should satisfy the following condition, in order to have a stable consensus filter.

$$\delta_{j} \leq \frac{1}{\deg_{\max}} \tag{19}$$

where $\deg_{\max}$ is the maximum degree of nodes for a specific network. In the maximization step (M-step), the global distribution parameter set is obtained at each observer by using the following equations:

$$\boldsymbol{\mu}_{k,c}^{q+1} = \frac{\boldsymbol{\chi}_{k,j,c}^{q+1}(2)}{N\boldsymbol{\chi}_{k,j,c}^{q+1}(1)}, \quad \boldsymbol{\Sigma}_{k,c}^{q+1} = \frac{\boldsymbol{\chi}_{k,j,c}^{q+1}(3)}{N\boldsymbol{\chi}_{k,j,c}^{q+1}(1)} \tag{20}$$

4. Proposed distributed MTT algorithm

In this section we present and discuss the block-diagram of our proposed distributed MTT algorithm. Starting with the initial marginal distributions for targets at each observer, the particles at current time step are generated from the proposal distribution, as of equation (8). Then the distributed EM algorithm is performed for sufficient iterations (q times) at each node and the global distribution set at the current time step is calculate and saved for each observer. In continue the new particle set is generated from the calculated GMM distributions for each target. The SMC-JPDAF algorithm with gating procedure is executed in the next step at each network node to obtain the new sample set with the corresponding weights for each target that is approximately distributed according to the marginal posterior distribution of the corresponding target. This procedure precedes recursively for the consecutive time steps. The advantages of the proposed distributed MTT algorithm may be discussed form two major aspects. First, the main advantage of the proposed distributed MTT algorithm is that, the approximations for the marginal posterior distributions of interest is available at each of the sensor nodes, i.e. the tracking results of networks nodes converge to each other by means of the consensus filter used. Furthermore, tracking performance would improve by increasing the number of sensor nodes. This is due to the spatial diversity obtained by spreading the sensors in the area of interest. So, the proposed algorithm is robust to the failure of some nodes, i.e. the algorithm has the ability to track the multiple targets, even with a few number of the sensor nodes. This feature is desired in military applications, where the high tracking performance and robustness against nodes failure are of interest.

As the second aspect, the advantage of the proposed algorithm is that, it has the ability to use sensor nodes with low processing ability as the network nodes and still maintain the desired tracking performance. In centralized implementation of MTT, large number of particles is needed to reach a desired tracking performance, leading to have sensor nodes with higher processing ability and so higher energy consumption. However, as it is shown in the next section, the proposed distributed MTT algorithm could reach the same tracking performance by rather a fewer number of particles for each observer. In other words, the tracking task will distribute in whole networks nodes.

This is the case of interest in implementation of low-cost sensor networks with relatively high tracking ability and reliability.

5. Simulation results

In this section the simulation results of our proposed distributed MTT system in a given small networks are obtained. For the sake of comparison, the simulation results of a centralized MTT system over the same network are given. In what follows all location positions and distance measures are in meters, angles in radians, time in seconds and velocity measures in meters per second. Also the near constant velocity model is used with standard deviation for system noises as $\sigma_x = \sigma_y = 0.1$ and for measurements noises as $\sigma_R = 5$ and $\sigma_\theta = 0.05$. The discrete time T is set to 1 second. The simulations are performed for two target case, i.e. $K = 2$. So the initial state vectors for two targets are assumed to be $[-50\ 1\ 50\ 1.5]$ and $[-50\ 1\ 0\ -0.5]$. The detection probability p_D for each target is set to 0.8 and the rate parameter for the Poisson distribution of the clutter measurements is assumed to be $\lambda_C = 0.5$. The parameter ε for the gating procedure as defined in [3] is empirically set to 40 and the maximum sensing range for observers is set to 100m. As the centralized MTT system, the SMC-JPDAF algorithm followed by the gating procedure is simulated for five observers in the network given in Fig. 1. The true track of two targets (solid lines) and the estimated track (colored dashed lines) by using this algorithm are plotted in Fig. 2. In this figure, the locations of observers are plotted as magenta circles and the number of particles is set to 100. In the proposed distributed MTT system, considering the maximum detection range of 100m for each of the observers leads to a connected network as shown in Fig. 3. The number of GMM components is set to 3 and the number of iterations of the distributed GMM algorithm in each time step is set to 10. Also the step size parameter for the consensus filter is empirically set to 0.005.

The true and estimated tracks of the targets using the proposed algorithm for 2 sample observer are depicted in figures 4-A to B. As it is obvious from figures 2 and 4, the tracking performance of the proposed algorithm for the given number of particles is better than the centralized algorithm. Also it is seen that the estimated track and tracking performance of the network nodes in the proposed system are nearly the same. So, the tracking results are available from each of nodes and this is a great advantage for a target tracking network.

Also the RMSE curves versus the number of particles for two systems are plotted in Fig. 5 and the curves for execution time of a single iteration of the program in central node for the centralized system and in each node in distributed system, versus the number of particles, are plotted in Fig. 6. As it is obvious from the above mentioned figures, in both cases the RMSE decreases and so the algorithm performance increases as the number of particles increases.

Comparing the RMSE curves for the centralized and the proposed distributed systems, the better tracking performance of the proposed system is obvious. Also the execution time of the proposed system is considerably less than the centralized one as shown in Fig. 6. This will lead to significantly increase in nodes' and also the network's life time. By more carefully inspecting figures 5 and 6 it could be found out that the proposed algorithm can reach a desired value for RMSE by rather a fewer number of particles leading to execution time less than the centralized algorithm. Furthermore the tracking performance of the proposed algorithm is less dependent of nodes failure, i.e., one node failure only means reduction of one of adjacent nodes for the neighbors of that node in whole network and this will negligibly reduce the tracking performance of the network especially for larger networks.

6. Conclusion Remarks

In this paper a kind of distributed multi-target tracking algorithm is proposed that could be used in arbitrary large and dense wireless sensor networks. The tracking system has the ability to track any general state space model. In order to jointly solve the tracking and data association problems, the simplified version of the Monte Carlo implementation of well-known JPDAF algorithm is used in network nodes. Also to make distributive inferences in whole network, the distributed implementation of the GMM algorithm is applied. The results are proved by simulation results where comparisons are made by the centralized algorithm under the same conditions.

7. Acknowledgement

This work was partially supported by Iran Telecommunication Research Center.

8. References

[1] S. J. Julier and J. K. Uhlmann, A new extension of the Kalman filter to nonlinear systems, in Proceedings of AeroSense: The 11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls, vol.

Multi Sensor Fusion, Tracking and Resource Management II, 1997.

[2] D. Schuls, W. Burgard, D. Fox, People tracking with mobile robots using sample-based joint probabilistic data association filters, International Journal of Robotics Research, vol. 22, no. 2, 2003.

[3] J. Vermaak, S. J. Godsill, and P. Perez, Monte Carlo filtering for multi-target tracking and data association, IEEE Transactions on Aerospace and Electronic Systems, vol. 41, no. 1, Jan. 2005, pp. 309-322.

[4] M. J. Coats, Distributed particle filtering for sensor networks, in Proc. of Int. Symp. Information Processing in Sensor Networks (IPSA), Berkeley, CA, Apr. 2004.

[5] L. Zuo, K. Mehrotra, P. Varshney, and C. Mohan, Band-width efficient target tracking in distributed sensor networks using particle filters, in Proc. of 14th European Signal Processing Conf. EURASIP2006, Florence, Italy, Sep. 2006.

[6] D. Gu, Distributed EM algorithm for Gaussian mixtures in sensor networks, accepted to be published in IEEE Transactions on Neural Networks, 2008.

[7] Y. Bar-Shalom, T. E. Fortmann, Tracking and data association, Academic Press, 1988.

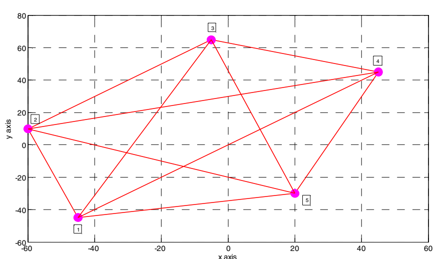

Figure 1. Network topology for centralized case. Solid lines are communication links.

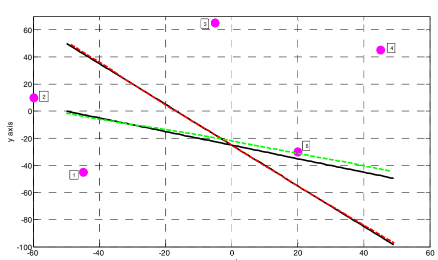

Figure 2. The true and estimated tracks using centralized algorithm.

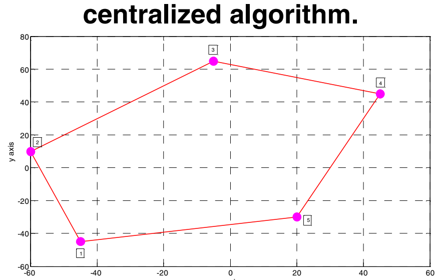

Figure 3. Network for proposed algorithm. Solid lines are communication links.

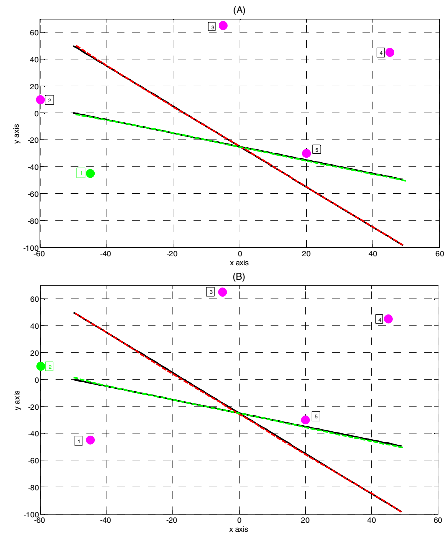

Figure 4. The true and estimated tracks using proposed algorithm for 2 sample observer.

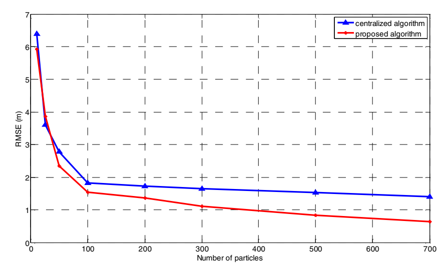

Figure 5. RMSE curves for the centralized and the proposed algorithm.

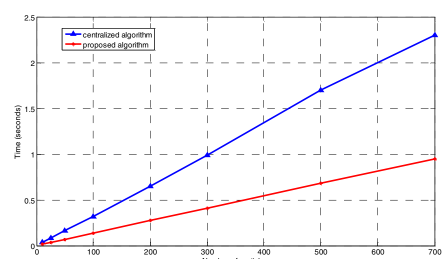

Figure 6. Time of execution for the centralized and the proposed algorithms.