PARTICLE FILTERING FOR TARGET TRACKING WITH MOBILE SENSORS

Yao Li and Petar M. Djurić

Department of Electrical and Computer Engineering
Stony Brook University
Stony Brook, NY, 11794, USA

ABSTRACT

Recent progress in distributed robotics and low power embedded systems has led to development of mobile sensor networks. Controlled mobility, moving sensors intentionally, enables a new set of possibilities in wireless sensor networks and facilitates many applications in signal processing areas such as target tracking. In this paper we consider the problem of tracking a target using three mobile sensors that measure the received signal strength (RSS) from the target. We propose the use of particle filtering where the positioning of the mobile sensor is based on the predicted target's positions. In deciding how to deploy the sensors, we have used the Cramér-Rao lower bound (CRLB) that we have derived for our scheme. The performance of the method is investigated by simulations and compared to tracking by traditional static sensor network.

Index Terms— wireless sensor networks, particle filtering, Posterior Cramér-Rao lower bound, Monte Carlo methods, root mean square error

1. INTRODUCTION

The rapid progress in micro-electro-mechanical systems (MEMS) and radio frequency (RF) design makes the development of wireless sensor networks (WSNs) with a variety of applications in both civil and military aspects of our human life quite intriguing. These networks involve large scale, low power, distributed sensors nodes with limited individual capability. The maximization of the sensing utility while increasing the effectiveness of the whole network has become a very popular research field nowadays [1].

The progress in distributed robotics and low power embedded systems has led to the development of mobile sensor networks [2]. Controlled mobility, moving sensors intentionally, enables a new set of possibilities for sensor networks. Mobile sensors often provide better information. In this paper we address networks with a fusion center, which in principle can perform tasks like detection, localization and tracking based on sensor measurements more accurately if the measurements come from mobile sensors [3].

In this paper we focus on the problem of target tracking with mobile sensors. Sensors trajectory planning is a critical issue for this problem. Observer trajectory planning has been addressed in [4], where the optimal paths are derived by maximizing the mutual information between the final target state and the entire measurement sequence. In [5], posterior Cramér-Rao lower bound (PCRLB) was used as the sensor motion criterion. These methods, however, involve a multi-step planning and impose strong memory requirements and computational burden to the system. Here we propose a recursive update of the sensors' positions based on online target state estimates obtained by a particle filtering algorithm [6]. At each time instant, the fusion center predicts the next position of the target, and based on it decides where to send the mobile sensors whose measurements are used for tracking the target.

This work has been supported by the National Science Foundation under CCF-0515246 and the Office of Naval Research under Award N00014-06-1-0012.

The paper is organized as follows: In Section 2, we formulate the problem, and in Section 3 we present the particle filtering algorithm with its implementation in our proposed mobile sensor network. We derive the PCRLB for mobile tracking in Section 4. In Section 5, we show the performance of the proposed algorithm with comparison to static sensor tracking. We make our final remarks in Sections 6.

2. PROBLEM FORMULATION

The target motion model [7] is based on the assumption that its velocity is subject to an unknown acceleration, which gives

$$\mathbf{x}_{t+1} = \mathbf{\Phi}_t \mathbf{x}_t + \mathbf{\Gamma}_t \mathbf{w}_t \tag{1}$$

where the transition matrix $\mathbf{\Phi}_t$ and $\mathbf{\Gamma}_t$ are given by

$$\mathbf{\Phi}_t = \begin{pmatrix} 1 & T_s \\ 0 & 1 \end{pmatrix} \otimes \mathbf{I}, \quad \mathbf{\Gamma}_t = \begin{pmatrix} \frac{T_s^2}{2} \\ T_s \end{pmatrix} \otimes \mathbf{I},$$

where $\otimes$ denotes the Kronecker product. The state vector for the target is defined by $\mathbf{x}_t = [x_{1,t}, x_{2,t}, \dot{x}_{1,t}, \dot{x}_{2,t}]^\top$, where $x_{1,t}, x_{2,t}$ denote the target location and $\dot{x}_{1,t}, \dot{x}_{2,t}$ the target velocity in a two-dimensional plane at time instant t. The symbol T_s is the sampling time interval and $\mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_w)$ is the target propagation noise, representing the target acceleration uncertainty. We assume that the target motions in the x_1 and x_2 directions are statistically independent, and thus $\mathbf{Q}_w = diag(\sigma_{w,1}^2, \sigma_{w,2}^2)$.

Sensor measurements could include time of arrival (TOA), direction of arrival (DOA), or received signal strength (RSS) [8]. Here we use the RSS as sensor observations. The RSS is comparatively less costly yet traditionally seen as a coarse measurement of range. The signal power reaching the $n-$th sensor can be measured as a log-normal variable approximated by a Gaussian distribution, i.e.,

$$y_t^{(n)} = P_0 - 10\alpha \log_{10}\left(\frac{|\mathbf{s}_t^{(n)} - \boldsymbol{\rho}_t|}{d_0}\right) + \varepsilon_t^{(n)} \tag{2}$$

where $\mathbf{s}_t^{(n)} = [s_{1,t}^{(n)}, s_{2,t}^{(n)}]$ and $\boldsymbol{\rho}_t = [x_{1,t}, x_{2,t}]$, for $n = 1, \cdots N$ are the positions of the n-th sensor and the target at time instant t, respectively; $\varepsilon_t^{(n)}$ is the measurement Gaussian noise with $\varepsilon_t^{(n)} \sim N(0, \sigma_\varepsilon^2)$; $P_0(dB)$ is the received power at a reference distance d_0; α is a parameter that is used to model path loss. The power P_0 and the paremetr α are assumed known.

The measured information, $y_t^{(1:N)}$, is sent to the fusion center at each time instant t. The objective is to track the target state $\mathbf{x}_{0:t}$ based on the observations $\mathbf{y}_{1:t} = y_{1:t}^{(1:N)}$.

3. PARTICLE FILTERING

Due to the nonlinearity in the measurement model, we apply particle filtering for target tracking [6]. We use Bayesian state estimation and there the information about the state vector given the observations is obtained from the *a posteriori* probability density function, $p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$. With particle filtering we approximate sequentially the posterior distributions using a set of random particles with their associated weights. The whole method consists of three main steps: (1) particle propagation,(2) weight updates followed by normalization, and (3) resampling.

If the random measure at time instant t is denoted by $\chi_t = \{\mathbf{x}_t^{(m)}, \omega_t^{(m)}\}_{m=1}^M$, the proposed particles are drawn from the importance function $\pi(\cdot)$, i.e.,

$$\mathbf{x}_t^{(m)} \sim \pi(\mathbf{x}_t|\mathbf{x}_{0:t-1}, \mathbf{y}_{1:t}) \tag{3}$$

for $m = 1, 2, ..., M$, where M is the number of particles. The calculation of the weight for each particle is followed by normalization, that is,

$$\omega_t^{*(m)} \propto \frac{p(\mathbf{y}_t|\mathbf{x}_t^{(m)})p(\mathbf{x}_t^{(m)}|\mathbf{x}_{t-1}^{(m)})}{\pi(\mathbf{x}_t^{(m)}|\mathbf{x}_{0:t-1}^{(m)}, \mathbf{y}_{1:t})}\omega_{t-1}^{*(m)} \tag{4}$$

and

$$\omega_t^{(m)} = \frac{\omega_t^{*(m)}}{\sum_{i=1}^M \omega_t^{*(i)}}. \tag{5}$$

The posterior density is approximated as

$$p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}) \approx \sum_{m=1}^M \omega_t^{(m)}\delta(\mathbf{x}_{0:t} - \mathbf{x}_{0:t}^{(m)}). \tag{6}$$

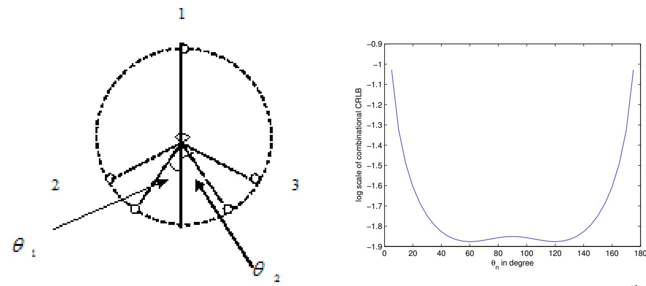
Resampling is performed whenever necessary in order to avoid the algorithm's degeneracy.

Here we do not describe the details of the implementation of particle filtering to static sensor networks because it has been well documented in the literature. Instead, we immediately address the tracking by particle filtering using mobile sensor measurements.

3.1. Proposed mobile tracking algorithm

Good performance of particle filtering using measurements of static sensors requires dense deployment of the sensors over the whole region of interest. If the trajectory of the target is in a broad area, this may need a very large number of sensors to be deployed. Large number of sensors may further translate to heavy computations at the fusion center. Moreover, the limited wireless channel capacity may impose problems that arise from simultaneous transmission of many signals. Here we propose a tracking methodology with mobile sensors that relies on a very small number of sensors and will not have such problems.

First we want to establish a strategy for sensor deployment. For simplicity, we use only three mobile sensors positioned on a circle in a two-dimensional plane. Assuming the target is at the center of the circle, without loss of generality, we fix one sensor and want to check the performance of localization of the target by varying the location of the other two sensors around the circle. To that end, we use the Cramér-Rao Lower Bound for target localization. We introduce the constraint that the other two sensors have symmetric positions with respect to the first sensor. In other words, the sensors are positioned as in Fig. 1(a) where we see them in two different positions. As we vary the angle θ from $0°$ to $180°$, the CRLB of the target location varies, achieving its minimum value for $\theta = 60°(120°)$, Fig 1(b).



(a) (b)

Fig. 1. (a) A target and three sensors deployed on a circle around the target. (b) CRLB of localization as a function of θ.

3.1.1. *Ideal Case*

The proposed tracking algorithm dynamically updates the locations of the mobile sensors by using the target state estimate obtained by particle filtering. At each time instant t, the fusion center performs estimation based on the information provided by the sensors. Given the estimated target state $\hat{\mathbf{x}}_t = [\hat{\boldsymbol{\rho}}_t, \hat{\mathbf{v}}_t]^\top$ and the motion model of the target, the fusion center predicts the target location at the next time instant $\tilde{\boldsymbol{\rho}}_{t+1}$, where

$$\tilde{\boldsymbol{\rho}}_{t+1} = \hat{\boldsymbol{\rho}}_t + \hat{\mathbf{v}}_t T_s. \tag{7}$$

The sensor positions at $t+1$ are assigned equidistantly around $\tilde{\boldsymbol{\rho}}_{t+1}$ on a circle with radius r. In the ideal case, the sensors are capable enough of moving to the designated places along with the target. The ideal location of the $n-$th sensor at time $t+1$ is given by

$$\tilde{s}_{1,t+1}^{(n)} = \tilde{x}_{1,t+1} - r\cos\theta_n \tag{8}$$

$$\tilde{s}_{2,t+1}^{(n)} = \tilde{x}_{2,t+1} - r\sin\theta_n \tag{9}$$

where $\theta_n = \frac{2\pi n}{N}$, $n = 1, ..., N$, and N is the number of sensors.

3.1.2. *Realistic Case*

In a more realistic scenario, we have to take into consideration the limitations in mobility of the sensors. We assume that the sensors follow a linear model for their motion, i.e.,

$$\begin{aligned} s_{1,t+1}^{(n)} &= s_{1,t}^{(n)} + u_{1,t}^{(n)}T_s \\ s_{2,t+1}^{(n)} &= s_{2,t}^{(n)} + u_{2,t}^{(n)}T_s \end{aligned} \tag{10}$$

where $u_{1,t}^{(n)} \in (u_{min_1}, u_{max_1})$ and $u_{2,t}^{(n)} \in (u_{min_2}, u_{max_2})$ denote the controlled velocity in the x_1 and x_2 directions, respectively, by the $n-$th sensor. The velocities are assumed constant from t to $t+1$. The choice of $[u_{1,t}^{(n)}, u_{2,t}^{(n)}]$ is subject to minimizing the cost function

$$f_{n,t+1} = ||\mathbf{s}_{t+1}^{(n)} - \tilde{\mathbf{s}}_{t+1}^{(n)}|| \tag{11}$$

where $\tilde{\mathbf{s}}_{t+1}^{(n)}$ is defined by (8) and (9) and

$$[u_{1,t}^{(n)}, u_{2,t}^{(n)}] = \underset{u_{1,t},u_{2,t}}{\operatorname{argmin}} f_{n,t+1}(\mathbf{s}_{t+1}^{(n)}, \tilde{\mathbf{s}}_{t+1}^{(n)}). \quad (12)$$

With the proposed method, the fusion center is capable in determining if the mobile sensors can or cannot continue with tracking. This is done based on the motion model and the sensors' mobility limitations.

4. POSTERIOR CRAMÉR-RAO LOWER BOUNDS

In this section we show the derivation of the PCRLB for target tracking with mobile sensor networks. The PCRLB provides the lower bound of the mean-square error (MSE) of the estimated states [9]. In the derivation, we follow the approach presented in [10].

First, we rewrite the target state vector in a block form $\mathbf{x}_t = [\mathbf{x}_t^{(1)}, \mathbf{x}_t^{(2)}]^\top$ where $\mathbf{x}_t^{(1)} = [v_{1,t}, v_{2,t}]^\top$ and $\mathbf{x}_t^{(2)} = [\rho_{1,t}, \rho_{2,t}]^\top$. The target motion model becomes

$$\mathbf{x}_{t+1}^{(1)} = \mathbf{x}_t^{(1)} + \mathbf{F}_t\mathbf{w}_t \quad (13)$$

$$\mathbf{x}_{t+1}^{(2)} = \mathbf{G}_t^{(1)}\mathbf{x}_t^{(1)} + \mathbf{G}_t^{(2)}\mathbf{x}_t^{(2)} + \mathbf{G}_t^{(3)}\mathbf{x}_{t+1}^{(1)} \quad (14)$$

where

$$\mathbf{F} = T_s I, \qquad \mathbf{G}_t^{(1)} = \frac{T_s}{2} I, \qquad \mathbf{G}_t^{(2)} = I, \qquad \mathbf{G}_t^{(3)} = \mathbf{G}_t^{(1)}.$$

The information matrix $\mathbf{J}_t$ can be updated recursively through

$$\mathbf{S}_{t+1} = \begin{bmatrix} \mathbf{S}_{t+1}^{11} & \mathbf{S}_{t+1}^{12} & \mathbf{S}_{t+1}^{13} \\ \mathbf{S}_{t+1}^{21} & \mathbf{S}_{t+1}^{22} & \mathbf{S}_{t+1}^{23} \\ \mathbf{S}_{t+1}^{31} & \mathbf{S}_{t+1}^{32} & \mathbf{S}_{t+1}^{33} \end{bmatrix} = \mathbf{M}_t^{-T} \begin{bmatrix} \mathbf{J}_t^{11} + \mathbf{H}_t^{11} & \mathbf{J}_t^{12} + \mathbf{H}_t^{12} & \mathbf{H}_t^{13} \\ (\mathbf{J}_t^{12} + \mathbf{H}_t^{12})^T & \mathbf{J}_t^{22} + \mathbf{H}_t^{22} & \mathbf{H}_t^{23} \\ (\mathbf{H}_t^{13})^T & (\mathbf{H}_t^{23})^T & \mathbf{H}_t^{33} \end{bmatrix} \mathbf{M}_t^{-1}$$

$$\mathbf{J}_{t+1} = \begin{bmatrix} \mathbf{S}_{t+1}^{22} & \mathbf{S}_{t+1}^{23} \\ \mathbf{S}_{t+1}^{32} & \mathbf{S}_{t+1}^{33} \end{bmatrix} - \begin{bmatrix} \mathbf{S}_{t+1}^{21} \\ \mathbf{S}_{t+1}^{31} \end{bmatrix} \left[\mathbf{S}_{t+1}^{11}\right]^{-1} \left[\mathbf{S}_{t+1}^{12} \quad \mathbf{S}_{t+1}^{13}\right] \quad (15)$$

where

$$\mathbf{J}_t = \begin{bmatrix} \mathbf{J}_t^{11} & \mathbf{J}_t^{12} \\ \mathbf{J}_t^{21} & \mathbf{J}_t^{22} \end{bmatrix}, \quad \mathbf{M}_t = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{G}_t^{(1)} & \mathbf{G}_t^{(2)} & \mathbf{G}_t^{(3)} \end{bmatrix}.$$

Here $\mathbf{S}_t$ is the information submatrix for $[\mathbf{x}_{t-1}^{(1)}, \mathbf{x}_t]$, and

$$\mathbf{H}_t^{11} = E\{-\Delta_{\mathbf{x}_t^{(1)}}^{\mathbf{x}_t^{(1)}} \log \bar{p}_t\}, \quad \mathbf{H}_t^{12} = E\{-\Delta_{\mathbf{x}_t^{(1)}}^{\mathbf{x}_t^{(2)}} \log \bar{p}_t\}$$

$$\mathbf{H}_t^{13} = E\{-\Delta_{\mathbf{x}_t^{(1)}}^{\mathbf{x}_{t+1}^{(1)}} \log \bar{p}_t\}, \quad \mathbf{H}_t^{22} = E\{-\Delta_{\mathbf{x}_t^{(2)}}^{\mathbf{x}_t^{(2)}} \log \bar{p}_t\}$$

$$\mathbf{H}_t^{23} = E\{-\Delta_{\mathbf{x}_t^{(2)}}^{\mathbf{x}_{t+1}^{(1)}} \log \bar{p}_t\}, \quad \mathbf{H}_t^{33} = E\{-\Delta_{\mathbf{x}_{t+1}^{(1)}}^{\mathbf{x}_{t+1}^{(1)}} \log \bar{p}_t\}$$

with

$$\bar{p}_t = p(\mathbf{x}_{t+1}^{(1)}|\mathbf{x}_t) \cdot p(\mathbf{y}_{t+1}|\mathbf{x}_t, \mathbf{x}_{t+1}^{(1)}). \quad (16)$$

We can write the $\mathbf{H}_t$ matrix as

$$\mathbf{H}_t = \mathbf{H}_{t,a} + \mathbf{H}_{t,b} \quad (17)$$

where

$$\mathbf{H}_{t,a} = E\{-\Delta \log p(\mathbf{x}_{t+1}^{(1)}|\mathbf{x}_t^{(1)})\}$$

$$\mathbf{H}_{t,b} = E\{-\Delta \log p(\mathbf{y}_{t+1}|\mathbf{x}_t, \mathbf{x}_{t+1}^{(1)}). \quad (18)$$

It can be shown that $\mathbf{H}_{t,a}^{11} = \operatorname{diag}\{\frac{1}{\sigma_{w,1}^2}, \frac{1}{\sigma_{w,2}^2}\}$ and

$$\mathbf{H}_{t,b}^{11} = E\{-\Delta_{\mathbf{v}_t}^{\mathbf{v}_t} \log p(\mathbf{y}_{t+1}|\mathbf{v}_t, \boldsymbol{\rho}_t, \mathbf{v}_{t+1})\}.$$

Solving for $\mathbf{H}_{t,b}^{11}$, we get

$$\mathbf{H}_{t,b}^{11} = A \cdot E\left[\sum_{n=1}^{N} \mathbf{B}_n^\top \mathbf{B}_n\right] \quad (19)$$

where

$$\mathbf{B}_n = \left[\frac{x_{1,t+1} - s_{1,t+1}^{(n)}}{|\boldsymbol{\rho}_{t+1} - \mathbf{s}_{t+1}^{(n)}|^2} \quad , \quad \frac{x_{2,t+1} - s_{2,t+1}^{(n)}}{|\boldsymbol{\rho}_{t+1} - \mathbf{s}_{t+1}^{(n)}|^2}\right]$$

and $A = (\frac{5T_s\alpha}{\sigma_\varepsilon \log 10})^2$. Recalling that the sensors positions are functions of the previous time target state estimates, they are also random variables. So $\mathbf{H}_{t,b}^{11}$ is the expectation over $\mathbf{v}_t$, $\mathbf{v}_{t+1}$, $\boldsymbol{\rho}_t$, $\mathbf{y}_{t+1}$ and $\mathbf{s}_{t+1}^{1:N}$. Under the considerations of symmetry and independence among the state space variables, we have

$$\mathbf{H}_{t,a}^{12} = \mathbf{H}_{t,a}^{22} = \mathbf{H}_{t,a}^{23} = \mathbf{0}$$

$$\mathbf{H}_{t,a}^{33} = \mathbf{H}_{t,a}^{11}, \qquad \mathbf{H}_{t,a}^{13} = -\mathbf{H}_{t,a}^{11}$$

$$\mathbf{H}_{t,b}^{13} = \mathbf{H}_{t,b}^{33} = \mathbf{H}_{t,b}^{11}$$

$$\mathbf{H}_{t,b}^{12} = \mathbf{H}_{t,b}^{23} = \frac{2}{T_s}\mathbf{H}_{t,b}^{11}, \qquad \mathbf{H}_{t,b}^{22} = \frac{2}{T_s}\mathbf{H}_{t,b}^{11}.$$

Given the results above, we can compute the filtering information matrix $\mathbf{J}_t$, whose diagonal elements are the PCRLBs of the unknown target states. Since the calculation of this PCRLB has no analytical solution in closed form, we resort to Monte Carlo simulation methods.

5. SIMULATION RESULTS

We carried out several computer simulations that compare the performance between our proposed tracking algorithm and the traditional static sensor tracking. For the static sensor network scenario, we considered $N = 50$ and $N = 100$ sensors deployed uniformly in the region. For the mobile tracking case, we applied $N = 3$ sensors only. The parameters in the sensor measurement model were set to $P_0 = 30(dB)$ at $d_0 = 1m$, and $\alpha = 2.3$. The background noise level was chosen as $\sigma_v = 4dB$.

The initialization state vector was assumed to have a Gaussian distribution with mean $\bar{\mathbf{x}}_0 = [0\ 0\ 5\ 5]$ and covariance matrix $\mathbf{C}_0 = \operatorname{diag}\{100^2\ \ 100^2\ \ 10^2\ \ 10^2\}$. The mobile sensors were positioned around the predicted location of the target on a circle with radius $r = 100$m. As a performance metric, we used the root mean square error (RMSE) of the estimated states. We ran the experiment with 50 independent trajectories and with 100 runs for each trajectory. The tracking time is 100 seconds with sampling interval of $T_s = 1$sec. We used 1000 particles. The results are shown in Figure 2. The same parameters were used in the computation of the PCRLBs. The results are shown in Figure 3. There we see that the mobile tracking algorithm achieves better performance with less than one tenth of the sensors from the static network. In Figures 4 and 5, we plotted the change of RMSE as function of the number of mobile sensors and the radius of the circle. As expected, they increased with less sensors and increased circle radius.

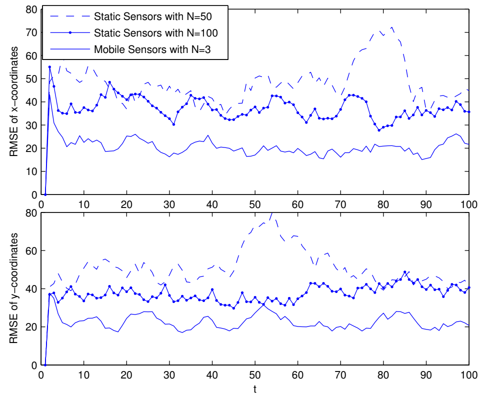


Fig. 2. Comparison for RMSE of $x_{1,t}$ and $x_{2,t}$.

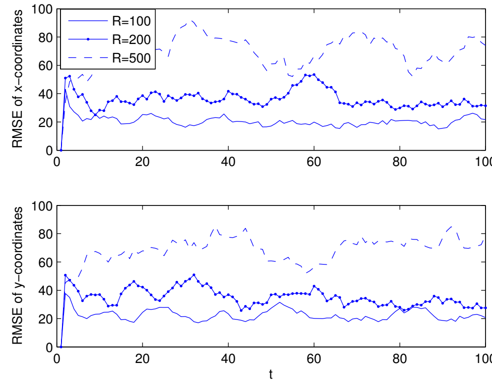


Fig. 4. RMSE for various radii.

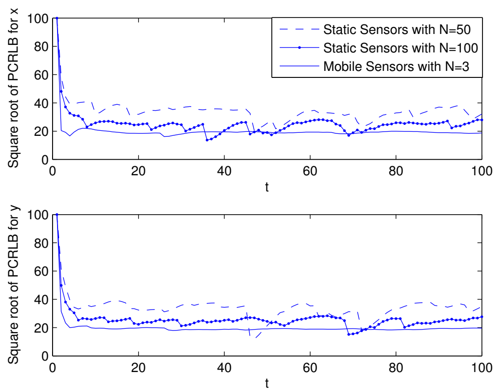


Fig. 3. Comparison of the square root of PCRLB of $x_{1,t}$ and $x_{2,t}$.

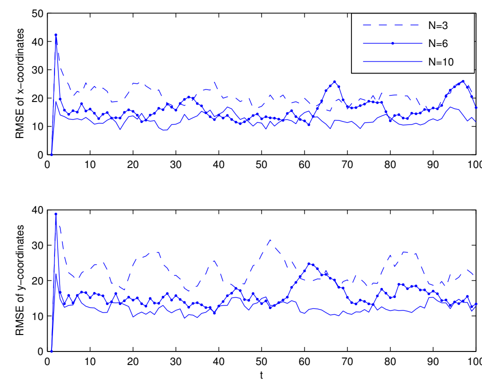


Fig. 5. RMSE for various sensors numbers

6. CONCLUSIONS

In this paper we proposed a tracking problem with a few number of mobile sensors using particle filtering algorithm. The sensors moving trajectories were designated based on the real time estimate of the target state. The one step ahead prediction methodology in planning the optimal sensors motion was investigated. We derived the Posterior Cramér-Rao lower Bounds for this tracking scenario. Some simulation results were presented comparing the performance of our mobile tracking algorithm with traditional static sensors tracking. It was shown that with far less number of sensors, the proposed method achieved better tracking results.

7. REFERENCES

[1] A. Logothetis and A. Isaksson, “On sensor scheduling via information theoretic criteria,” Proceedings. American Control Conference, 1999, vol. 4, pp. 2402–2406.

[2] K. Dantu, M. H. Rahimi, H. Shah, S. Babel, A. Dhariwal, and G. S. Sukhatme, “Robomote: enabling mobility in sensor networks,” Los Angeles, California, 2005.

[3] T.-L. Chin, P. Ramanathan, K. K. Saluja, and K.-C. Wang, “Exposure for collaborative detection using mobile sensor networks,” Nov 2005.

[4] A.Isaksson A.Logothetis and R.J.Evans, “An information theoretic approach to observer path design for bearings-only tracking,” Proceedings. Decision and Control Conference, 1997, vol. 4, pp. 3132–3137.

[5] M. Hernandez, “Optimal sensor trajectories in bearings-only tracking,” Proceedings. 7th international conference on information fusion, 2004, pp. 893–900.

[6] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking,” *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[7] N. Bergman U. Forssel J. Jansson R. Karlsson F. Gustaffson, F. Gunnarsson and P. J. Nordlund, “Particle filtering for positioning, navigation, and tracking,” *IEEE Transactions on Signal Processing*, pp. 425–437, Feb 2002.

[8] M. Perkins N. S. Correal N. Patwari, A. O. Hero and R. J. ODea, “Relative location estimation in wireless sensor networks,” *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, vol. 51, no. 8, AUGUST 2003.

[9] H. L. Van Trees, *Detection, Estimaton, and Modulation Theory*, vol. 1, Wiley, New York, 1968.

[10] P. Tichavsky, C. H. Muravchik, and A. Nehorai, “Posterior cramer-rao bounds for discrete-time nonlinear filtering,” *IEEE Transactions on Signal Processing*, vol. 46, no. 5, May 1998.