TARGET TRACKING WITH MOBILE SENSORS USING COST-REFERENCE PARTICLE FILTERING

Yao Li and Petar M. Djurić

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

ABSTRACT

Sequential Monte Carlo (SMC) methods, also referred to as particle filters, have been successfully applied to a variety of highly nonlinear problems such as target tracking with sensor networks. In this paper, we propose the application of a new class of SMC methods named cost-reference particle filters (CRPFs) to target tracking with mobile sensors. CRPF techniques have been shown to be a flexible and robust alternative when there is no knowledge about the probability distributions of the noise in the system. The sensors positioning during tracking is determined by the predicted target's location as obtained by the CRPF. The performance of the method is investigated by simulations and compared to tracking with standard particle filters (SPFs).

***Index Terms*—** sequential Monte Carlo methods, cost-reference particle filters, mobile sensor network.

1. INTRODUCTION

Mobile sensor networks have undergone through significant developments due to the rapid progress in distributed robotics and low power embedded systems. Controlled mobility of sensors enables a new set of possibilities including target tracking with mobile sensors. For such tasks, sensors trajectory planning has been addressed as one of the most important problems. The criteria applied for movement of the sensors include maximizing the mutual information between the final target state and the measurement sequence [1] and minimizing the posterior Cramér-Rao lower bound (PCRLB) [2]. These methods involve a multi-step planning which imposes strong memory requirements and computational burden to the system. In [3], an online recursive update of sensors' positions based on real time target state estimate by standard particle filters (SPFs) was introduced. Yet this relies on the right assumptions of system noise distribution, which remains unknown in many situations.

In this paper, we introduce the application of a new class of particle filters, also known as Cost-Reference Particle Filters (CRPFs) [4], to the target tracking with mobile sensors. The main feature of the new method is that it is not based on any particular assumptions of the distribution of the system processing noise. Instead, a user-defined cost function is applied measuring the quality of the state signal estimates using the available observations. The implementation of our scheme is based on a centralized strategy, that is, a fusion center receives all the sensed information, processes it and makes all the decisions.

The paper is organized as follows: in Section 2 we present the fundamentals of the CRPF algorithm. In Section 3, the CRPF-based tracking implementation in our mobile sensor network is introduced. We show the tracking performance of the CRPF and compare it with that of the SPF in Section 4. Finally, some concluding remarks are made in Section 5.

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.

2. COST REFERENCE PARTICLE FILTERING

The discrete-time dynamic system used in many signal processing applications can be described as

$$\begin{aligned} \mathbf{x}_t &= f_x(\mathbf{x}_{t-1}) + \mathbf{w}_t \\ \mathbf{y}_t &= f_y(\mathbf{x}_t) + \boldsymbol{\epsilon}_t \end{aligned} \qquad (1)$$

where $\mathbf{x}_t$ is the system state vector at time t, $f_x(\cdot)$ is the state transition function, $\mathbf{w}_t$ is the state perturbation at time t, $\mathbf{y}_t$ denotes the observation vector at time t, $f_y(\cdot)$ is the measurement function transforming the state, and $\boldsymbol{\epsilon}_t$ represents the observation noise vector at time t. In SPF [5] where the probability distributions of the noise processes are known, the filtering problem is aimed at the online estimation of the *a posteriori* probability density function, $p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$. Yet in many situations when the processing noise is unknown, SPF still relies on the assumptions of knowing these probability distributions, which eventually may lead to quite inaccurate estimation results when the assumptions are incorrect .

Recently, a new class of particle filters called CRPFs has been proposed to address the filtering problem when there is no knowledge of the distributions of the system noises except for their means (without loss of generality, we assume that the noises have zero means) [4]. In CRPF methods, a user-defined cost and risk functions are used for assigning particle weights, which are then used for resampling. At time instant t, the random measure of the system state is denoted by $\chi_t =$

$\{\mathbf{x}_t^{(m)}, \mathcal{C}_t^{(m)}\}_{m=1}^M$ where $\mathcal{C}_t^{(m)}$ is the cost of the mth particle and M is the total number of particles. The measure χ_t is recursively updated upon the reception of the measurements $\mathbf{y}_t$. The implementation of CRPF is described as follows:

1. At time $t+1$, resampling takes place according to the probability mass function (pmf) $\hat{\pi}_{t+1}^{(m)}$ for $m = 1, 2, \ldots, M$. Here $\hat{\pi}_{t+1}^{(m)}$ is dependant on the one-step risk $\mathcal{R}_{t+1}^{(m)}$ of particle m which is defined as:

$$\mathcal{R}_{t+1}^{(m)} = \lambda \mathcal{C}_t^{(m)} + \mathcal{R}(\mathbf{x}_t^{(m)}|\mathbf{y}_{t+1}) \tag{2}$$

and

$$\hat{\pi}_{t+1}^{(m)} \propto \mu(\mathcal{R}_{t+1}^{(m)}) \tag{3}$$

where $\mu(\cdot)$ is a monotonically decreasing function, and $0 \leq \lambda \leq 1$ is a forgetting factor that prevents assignment of excessive weights to previous observations. The risk function $\mathcal{R}(\mathbf{x}_t|\mathbf{y}_{t+1})$ could be viewed as a prediction of the cost increment $\Delta\mathcal{C}(\mathbf{x}_{t+1}|\mathbf{y}_{t+1})$ that can be obtained before $\mathbf{x}_t$ is actually propagated. The new measure after resampling is denoted as $\hat{\chi}_t = \{\hat{\mathbf{x}}_t^{(m)}, \hat{\mathcal{C}}_t^{(m)}\}_{m=1}^M$.

2. Random particles $\mathbf{x}_{t+1}^{(m)}$ for $m = 1, \ldots, M$, are propagated by employing a user defined proposal density $p_{t+1}(\mathbf{x}_{t+1}| \hat{\mathbf{x}}_t^{(m)})$ which satisfies

$$E_{p_{t+1}(\mathbf{x}_{t+1}|\hat{\mathbf{x}}_t^{(m)})}[\mathbf{x}_{t+1}] = f_x(\hat{\mathbf{x}}_t^{(m)}) \tag{4}$$

where $E_p[\cdot]$ denotes the expectation with respect to the pdf in the subindex. An easy way to implement the selection of this function is to use a Gaussian kernel with adaptively selected variance [4].

3. The particle costs are updated as follows:

$$\mathcal{C}_{t+1}^{(m)} = \lambda \mathcal{C}_t^{(m)} + \Delta\mathcal{C}(\mathbf{x}_{t+1}^{(m)}|\mathbf{y}_{t+1}) \tag{5}$$

The incremental cost function $\Delta\mathcal{C}(\mathbf{x}_{t+1}^{(m)}|\mathbf{y}_{t+1})$ can have a simple form such as

$$\Delta\mathcal{C}(\mathbf{x}_{t+1}^{(m)}|\mathbf{y}_{t+1}) = \|\mathbf{y}_{t+1} - f_y(\mathbf{x}_{t+1}^{(m)})\|^q \tag{6}$$

where $q \geq 1$.

4. The state estimation can be obtained by

$$\hat{\mathbf{x}}_{t+1} = \sum_{m=1}^{M} \mathbf{x}_{t+1}^{(m)} \pi_{t+1}^{(m)} \tag{7}$$

where $\pi_{t+1}^{(m)} \propto \mu(\mathcal{C}_{t+1}^{(m)})$, for $m = 1, \ldots, M$.

3. TRACKING WITH MOBILE SENSORS

In [3], a tracking methodology was proposed with mobile sensors, where the sensors' locations are recursively updated by an online target state estimation. SPF algorithm was applied with prior knowledge of the system noise distributions. In this section, we implement for tracking a CRPF algorithm, where no assumptions about the system noise distributions are made.

The target motion model [6] is assumed to be subject to an unknown acceleration expressed by

$$\mathbf{x}_{t+1} = \mathbf{\Phi}_t \mathbf{x}_t + \mathbf{\Gamma}_t \mathbf{w}_t \tag{8}$$

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

$$\mathbf{\Phi}_t = \begin{pmatrix} 1 & T_s \\ 0 & 1 \end{pmatrix} \otimes \mathbf{I_2}, \quad \mathbf{\Gamma}_t = \begin{pmatrix} \frac{T_s^2}{2} \\ T_s \end{pmatrix} \otimes \mathbf{I_2},$$

where $\otimes$ denotes 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 $\mathbf{w}_t$ 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.

The observations are obtained by received signal strength-based (RSS) sensors [7]. The signal power reaching the $n-$th sensor can be measured as a random log-normal variable, i.e.,

$$y_t^{(n)} = P_0 - 10\alpha \log_{10}\left(\frac{|\mathbf{s}_t^{(n)} - \boldsymbol{\rho}_t|}{d_0}\right) + \varepsilon_t^{(n)} \tag{9}$$

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 noise with unknown distribution; $P_0(dB)$ 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 parameter α are assumed known. The measured information, $y_t^{(1:N)}$, is sent to a fusion center at each time instant t and the objective is to track the target state $\mathbf{x}_{0:t}$ based on the observations $\mathbf{y}_{1:t} = y_{1:t}^{(1:N)}$.

The sensors are assumed moving by following the linear model

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

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

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

and $\tilde{\mathbf{s}}_{t+1}^{(n)}$ is the projected sensor's positions symmetrically circled around the predicted target location $\tilde{\boldsymbol{\rho}}_{t+1}$ [3], where

$$\tilde{\boldsymbol{\rho}}_{t+1} = \hat{\boldsymbol{\rho}}_t + \hat{\mathbf{v}}_t T_s. \tag{12}$$

The whole tracking procedure is summarized in Table 1.

4. SIMULATION RESULTS

In this section we investigate the performance of the CRPF-based tracking algorithm and compare it with the performance of a method that uses the traditional SPF. The tracking was based on the measurements of $N = 3$ sensors. The parameters in the sensor measurement model were set to $P_0 = 30(dB)$ at $d_0 = 1m$, and $\alpha = 2.3$. The observation noise was modelled as a mixture Gaussian vector

$$\varepsilon_t^{(n)} \sim 0.1\mathcal{N}(0, 50) + 0.4\mathcal{N}(0, 10) + 0.5\mathcal{N}(0, 1)$$

for $n = 1, \ldots, N$.

The initialization state vector was assumed to have a Gaussian distribution with mean $\bar{\mathbf{x}}_0 = [0\ 0\ 1\ 1]$ and covariance matrix $\mathbf{C}_0 = \text{diag}\{25,\ 25,\ 1,\ 1\}$. The mobile sensors were positioned around the predicted location of the target on a circle with radius $r = 500m$. The state propagation noise was also modelled as a mixture Gaussian, where

$$w_{i,t} \sim 0.1\mathcal{N}(0, 1) + 0.5\mathcal{N}(0, 0.1) + 0.4\mathcal{N}(0, 0.01)$$

where $i = 1, 2$. In the CRPF-based tracking, the used cost and risk functions were

$$\begin{aligned} \mathcal{C}(\mathbf{x}_0) &= 0 \\ \Delta\mathcal{C}(\mathbf{x}_t|\mathbf{y}_t) &= \|\mathbf{y}_t - f_y(\mathbf{x}_t)\|^2 \\ \mathcal{R}(\mathbf{x}_t|\mathbf{y}_{t+1}) &= \|\mathbf{y}_{t+1} - f_y(\mathbf{\Phi}_t\mathbf{x}_t)\|^2. \end{aligned} \tag{13}$$

The mapping function μ that transforms the costs and risks into pmfs used for resampling and estimation had the following forms:

$$\begin{aligned} \mu_1(\mathcal{C}_t^{(m)}) &= \frac{1}{\mathcal{C}_t^{(m)}} \\ \mu_2(\mathcal{C}_t^{(m)}) &= \frac{1}{(\mathcal{C}_t^{(m)} - \min_i\{\mathcal{C}_t^{(i)}\} + \delta)^\beta} \end{aligned} \tag{14}$$

where $\delta = 0.01$ and $\beta = 2$, corresponding to CRPF1 and CRPF2 methods in the simulations. The forgetting factor was set to $\lambda = 0.95$. For particle propagation, we used a Gaussian density with adaptively selected covariance matrix $\mathbf{C}_{t+1}$, where $\mathbf{C}_{t+1} = \text{diag}\{\sigma_{1,t+1}^{2,(m)}, \sigma_{2,t+1}^{2,(m)}, \sigma_{3,t+1}^{2,(m)}, \sigma_{4,t+1}^{2,(m)}\}$ and

$$\sigma_{i,t}^{2,(m)} = \frac{t-1}{t}\sigma_{i,t-1}^{2,(m)} + \frac{\|\mathbf{x}_{i,t}^{(m)} - f_x(\hat{\mathbf{x}}_{i,t-1})^{(m)}\|^2}{td} \tag{15}$$

after $t > 10$, where $d = 4$ is the target state dimension. The initial values were $\sigma_{1:2,0}^{2,(m)} = 5$ for positions and $\sigma_{3:4,0}^{2,(m)} = 0.1$ for velocity, which were kept unchanged until $t = 10$.

We implemented the SPF-based tracking method with the true mixture Gaussian density and with a mismatched noise distribution. In the mismatched case, $\mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_w)$, where $\mathbf{Q}_w = \text{diag}\{\sigma_{w,1}^2, \sigma_{w,2}^2\}$ and $\sigma_{w,1} = \sigma_{w,2} = 0.1$; $\varepsilon_t^{(n)} \sim \mathcal{N}(0, 100)$. The tracking time was 800 seconds with a sampling interval of $T_s = 1$sec. The used number of particles was $M = 300$. Figure 1 shows the target trajectory in a single simulation run and the estimates corresponding to the SPF- and CRPF-based algorithms. As a performance metric, we applied the root mean square error (RMSE) of the estimates. We ran the experiment with 50 independent trajectories and the results are shown in Figures 2 and 3 for position and velocity, respectively . As can be seen from the graphs, the CRPF performed almost as well as the SPF method that uses correct information, and much better than the SPF that is based on incorrect distributional assumptions.

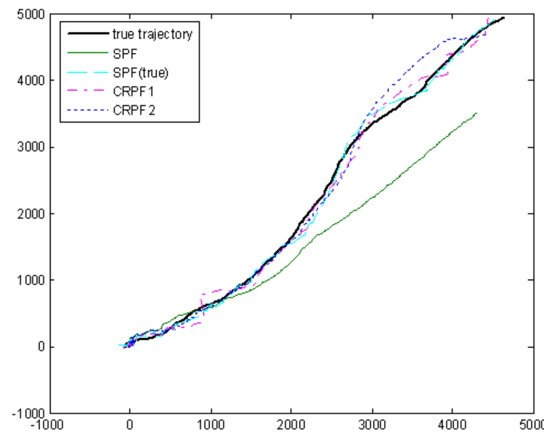

Fig. 1. target trajectory vs estimates

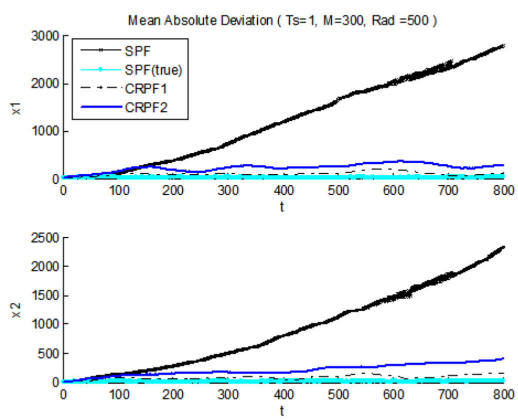

Fig. 2. RMSE comparison for position

5. CONCLUSIONS

In this paper we proposed tracking a target using RSS measurements of a few mobile sensors and based on cost-reference particle filtering. Being a more flexible and robust algorithm, the CRPF allows for estimation without prior knowledge of any system probability density functions. The computer simulation results demonstrate good performance of the CRPF.

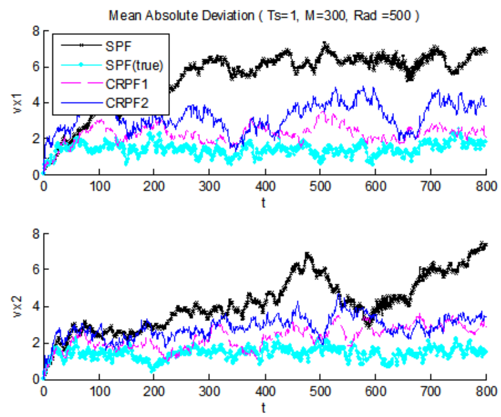

Fig. 3. RMSE comparison for velocity

6. REFERENCES

[1] A. Logothetis, A. Isaksson, 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.

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

[3] Y. Li and P. M. Djurić, "Particle filtering for target tracking with mobile sensors," IEEE International Conference on Acoustics, Speech and Signal Processing, Apr 2007.

[4] J. Míguez, M. F. Bugallo, and P. M. Djurić, "A new class of particle filters for random dynamic systems with unknown statistics," *EURASIP Journal on Applied Signal Processing*, vol. 15, pp. 2278–2294, 2004.

[5] 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.

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

[7] N. Patwari, A. O. Hero, M. Perkins, N. S. Correal, and R. J. O'Dea, "Relative location estimation in wireless sensor networks," *IEEE Transactions on Signal Processing*, vol. 51, no. 8, 2003.

Table 1. Tracking with mobile sensors

1. Initialize the algorithm
2. for $t = 1 \dots T$
   - Resample through $\hat{\pi}_{t+1}^{(m)}$, obtaining new particles sets as $\hat{\chi}_t = \{\hat{x}_t^{(m)}, \hat{\mathcal{C}}_t^{(m)}\}_{m=1}^M$

$$\begin{aligned}\mathcal{R}_{t+1}^{(m)} &= \lambda\mathcal{C}_t^{(m)} + \mathcal{R}(\mathbf{x}_t^{(m)}|\mathbf{y}_{t+1})\\ \hat{\pi}_{t+1}^{(m)} &\propto \mu(\mathcal{R}_{t+1}^{(m)})\end{aligned}$$

   - Propagate particles through proposal density $p_{t+1}(\mathbf{x}_{t+1}|\hat{\mathbf{x}}_t^{(m)})$
   - Update particle costs and normalization

$$\begin{aligned}\mathcal{C}_{t+1}^{(m)} &= \lambda\mathcal{C}_t^{(m)} + \Delta\mathcal{C}(\mathbf{x}_{t+1}^{(m)}|\mathbf{y}_{t+1})\\ \pi_{t+1}^{(m)} &\propto \mu(\mathcal{C}_{t+1}^{(m)})\\ \pi_{t+1}^{(m)} &= \frac{\pi_{t+1}^{(m)}}{\sum_{m=1}^{M}\pi_{t+1}^{(m)}}\end{aligned}$$

   for $m = 1, \dots, M$

   - Target state estimation

$$\hat{\mathbf{x}}_{t+1} = \sum_{m=1}^{M}\mathbf{x}_{t+1}^{(m)}\pi_{t+1}^{(m)}$$

   - Sensors Motion Control:
     - Calculate the projected sensor locations by

$$\begin{aligned}\tilde{s}_{1,t+2}^{(n)} &= \tilde{\rho}_{1,t+2} - r\cos\theta_n\\ \tilde{s}_{2,t+2}^{(n)} &= \tilde{\rho}_{2,t+2} - r\sin\theta_n\end{aligned}$$

     where $\theta_n = \frac{2\pi n}{N}$, for $n = 1, \cdots, N$, r is the circle radius and

$$\tilde{\boldsymbol{\rho}}_{t+2} = \hat{\boldsymbol{\rho}}_{t+1} + \hat{\mathbf{v}}_{t+1}T_s$$

     - Choose the control input for the sensors following

$$\mathbf{u}_{t+1}^{(n)} = \arg\min_{\mathbf{u}_{t+1}} ||\mathbf{s}_{t+2}^{(n)} - \tilde{\mathbf{s}}_{t+2}^{(n)}||$$

     - Move the sensors with corresponding control signals to

$$\mathbf{s}_{t+2}^{(n)} = \mathbf{s}_{t+1}^{(n)} + \mathbf{u}_{t+1}^{(n)}T_s$$

     for $n = 1, \dots, N$.