Tracking multiple targets with particle filtering using multiple receivers

C. Hue* J.-P. Le Cadre P. Pérez
IRISA/Université de Rennes1 IRISA/CNRS Microsoft Research
e-mail : {chue,lecadre}@irisa.fr pperez@microsoft.com
* Corresponding author

1 Introduction

Multitarget tracking (MTT) deals with the state estimation of an unknown number of moving targets. Available measurements may both arise from the targets if they are detected, and from clutter. Clutter is generally considered as a model describing false alarms. Its (spatio-temporal) statistical properties are quite different from those of the target, which makes the extraction of target tracks from clutter possible. To perform multitarget tracking the observer has at his disposal a huge amount of data, possibly collected on multiple receivers. Elementary measurements are receiver outputs, e.g., bearings, ranges, time-delays, Dopplers, etc. But the main difficulty comes from the assignment of a given measurement to a target model. These assignments are generally unknown, as are the true target models. Thus, two distinct problems have to be solved *jointly*: the data association and the estimation. We propose here a general algorithm for multitarget tracking in the passive sonar context and take advantage of its versatility to extend it to multiple receivers.

2 The basic particle filter

- Initialization: $\begin{cases} s_0^n \sim p(X_0) \\ q_0^n = 1/N \end{cases} \quad n = 1, \dots, N.$
- For $t = 1, \dots, T$:
  - Proposal: sample $\tilde{s}_t^n$ from $f(X_t | X_{t-1} = s_{t-1}^n, Y_t = y_t)$ for $n = 1, \dots, N$.
  - Weighting: $\begin{cases} \text{Compute un-normalised weights: } \tilde{q}_t^n = q_{t-1}^n \frac{p(\tilde{s}_t^n | s_{t-1}^n)\, l_t(y_t; \tilde{s}_t^n)}{f(\tilde{s}_t^n | s_{t-1}^n, y_t)} \text{ for } n = 1, \dots, N. \\ \text{Normalise weights: } q_t^n = \frac{\tilde{q}_t^n}{\sum_{n=1}^N \tilde{q}_t^n} \text{ for } n = 1, \dots, N. \end{cases}$
  - Return $\widehat{\mathbb{E}} g(X_t) = \sum_{n=1}^N q_t^n g(\tilde{s}_t^n)$.
  - Calculate $\hat{N}_{eff} = \frac{1}{\sum_{n=1}^N (q_t^n)^2}$.
  - Resampling: if $\hat{N}_{eff} < N_{threshold}$: $\begin{cases} s_t^n \sim \sum_{k=1}^N q_t^k \delta_{\tilde{s}_t^k} \\ q_t^n = 1/N \end{cases}$ else $s_t^n = \tilde{s}_t^n$ for $n = 1, \dots, N$.

Figure 1: Basic particle filter with adaptive resampling .

We consider a dynamic system represented by the stochastic process $(X_t) \in \mathbb{R}^{n_x}$ whose temporal evolution is given by the state equation (1). It is observed at discrete times via realizations of the stochastic process $(Y_t) \in \mathbb{R}^{n_y}$

governed by the measurement equation (2):

$$X_t = F_t(X_{t-1}, V_t), \tag{1}$$

$$Y_t = H_t(X_t, W_t). \tag{2}$$

The two processes $(V_t) \in \mathbb{R}^{n_v}$ and $(W_t) \in \mathbb{R}^{n_w}$ are only supposed to be independent white noises. Moreover, it is to be noted that no linearity hypothesis on F_t and H_t is done. We will denote by $Y_{0:t}$ the sequence of the random variables $(Y_0, \ldots, Y_t)$ and by $y_{0:t}$ one realization of this sequence. Our problem consists of computing at each time t the conditional density L_t of the state X_t given all the observations accumulated up to t, i.e., $L_t = p(X_t | Y_0 = y_0, \ldots, Y_t = y_t)$ and also of estimating any functional $g(X_t)$ of the state by the expectation $\mathbb{E}(g(X_t)|Y_{0:t})$. The original particle filter, named bootstrap filter [3], proposes to approximate the densities $(L_t)_t$ by a finite weighted sum of N Dirac densities centred on elements of $\mathbb{R}^{n_x}$, named particles. It then evolves the particle set $S_t = (s_t^n, q_t^n)_{n=1,\ldots,N}$, where $s_t \in \mathbb{R}^{n_x}$ is the particle and q_t its weight, such that the density L_t can be approximated by the density $L_{S_t} = \sum_{n=1}^{N} q_t^n \delta_{s_t^n}$. The two steps of prediction and correction are discretized as described in figure 1, and also the adaptive resampling step. The reader will find more details on the different filters in [3], [6] or [2] and on adaptive resampling in [7] and [2]. After these recalls, we propose an extension of this algorithm to multiple-object tracking.

3 Multitarget particle filter

3.1 Notations

Let M be the number of targets to track, assumed here to be known. The state vector we have to estimate is made by concatenating the state vector of each target. At time t, $X_t = (X_t^1, \ldots, X_t^M)$ follows the state equation (1) decomposed in M partial equations:

$$X_t^i = F_t^i(X_{t-1}^i, V_t^i) \qquad \forall i = 1, \ldots, M. \tag{3}$$

The noises (V_t^i) and $(V_t^{i'})$ are only supposed to be white both temporally and spatially, independent for $i \neq i'$. The observation vector at time t is denoted by $y_t = (y_t^1, \ldots, y_t^{m_t})$. Following the seminal ideas of R. Streit and T. Luginbuhl [9], we introduce the stochastic vector $K_t \in \{1, \ldots, M\}^{m_t}$ such that $K_t^j = i$ if y_t^j is issued from the i^{th} target. In this case, y_t^j is a realization of the stochastic process:

$$Y_t^j = H_t^i(X_t^i, W_t^j) \text{ if } K_t^j = i. \tag{4}$$

Again, the noises (W_t^j) and $(W_t^{j'})$ are only supposed to be white noises, independent for $j \neq j'$. We make the assumption that one measurement can originate from one target or from the clutter, and that one target can produce zero or several measurements at one time. For that, we dedicate the model 0 to false alarms. The false alarms are supposed to be uniformly distributed in the observation area. Their number is assumed to arise from a Poisson density of parameter λV where V is the volume of the observation area and λ the number of false alarms per volume unity. Of course, we do not associate any kinematic model to false alarms and then no particles represent their density. Let $\pi_t \in [0, 1]^{M+1}$ be defined by $\pi_t^i = \mathbb{P}(K_t^j = i)$ for all $j = 1, \ldots m_t$. It is assumed that the assignment vector K_t has independent components [9]. To estimate the density $L_t = p(X_t = (X_t^1, \ldots, X_t^M)|y_{0:t})$, we propose to use particles whose dimension is the sum of the ones of the individual state spaces corresponding to each target. Each of these concatenated vectors then gives jointly a representation of all targets. The proposal step is performed like in the single-target case. To update the weight of the particles, we need to estimate the association probabilities $(\pi_t^i)_{i=1,\ldots,M}$, which can be seen as the stochastic coefficients of the $M-$component mixture. To estimate them we propose to use a Gibbs sampler whose principles are briefly recalled (see [1] or [8] for more details). For $\theta_t = (X_t, K_t, \Pi_t)$, the method consists of generating a Markov chain that converges to the stationary distribution $p(\theta_t|Y_{0:t})$ which cannot be sampled directly. For that, we must be able to get a partition $\theta_t^1, \ldots, \theta_t^P$ of θ_t, and to sample alternatively from the conditional posterior distribution of each component of the partition. Assume the τ first elements of the Markov chain $(\theta_1, \ldots, \theta_\tau)$ have been drawn. We sample the P components of $\theta_{\tau+1}$ as follows:

- Initialization: $\begin{cases} s_0^n \sim p(X_0) \\ q_0^n = 1/N \end{cases} \quad n = 1, \dots, N.$
- For $t = 1, \dots, T$:
  - Proposal: sample $\tilde{s}_t^n$ from $f(X_t | X_{t-1} = s_{t-1}^n, Y_t = y_t)$ for $n = 1, \dots, N$.
  - Weighting:
    1. Initialization of the Gibbs sampler: $\begin{cases} \pi_{t,0}^i = \frac{1-\pi_t^0}{M} & i = 1, \dots, M; \\ X_{t,0}^i = \sum_{n=1}^N q_{t-1}^n \tilde{s}_t^{n,i} & i = 1, \dots, M. \end{cases}$
    2. For $\tau = 0, \dots, \tau_{end}$:

       a. $K_{t,\tau+1}^j \sim \; p(K_{t,\tau+1}^j = i) \propto \begin{cases} \pi_{t,\tau}^i l_t^i(y_t^j; x_{t,\tau}^i) & \text{if } i = 1, \dots, M; \\ \pi_t^0 / V & \text{if } i = 0. \end{cases}$

       b. $\pi_{t,\tau+1}^{1:M} \sim \mathcal{D}((1 + n^i(K_{t,\tau+1}))_{i=1,\dots,M})$, $n^i(K) \triangleq \sharp\{j : K^j = i\}$.

       c. For each i such that $\exists j^1, \dots, j^i / K_{t,\tau+1}^{j^l} = i$,

       $$\blacktriangleright \begin{cases} \sigma_{\tau+1}^n = \tilde{s}_t^{n,i} \\ \chi_{\tau+1}^n = \frac{p(y_t^{j^1}, \dots, y_t^{j^i} | X_t^i = \sigma_{\tau+1}^n) q_{t-1}^n}{\sum_{n=1}^N p(y_t^{j^1}, \dots, y_t^{j^i} | X_t^i = \sigma_{\tau+1}^n) q_{t-1}^n} \end{cases} \quad n = 1, \dots, N.$$

       $$\blacktriangleright X_{t,\tau+1}^i \sim \sum_{n=1}^N \chi_{\tau+1}^n \delta_{\sigma_{\tau+1}^n}$$

       d. For each i such that $\nexists j / K_t^j = i$, $X_{t,\tau+1}^i \sim \sum_{n=1}^N q_{t-1}^n \delta_{\tilde{s}_t^{n,i}}$.
    3. $\hat{\pi}_t^i = \frac{1}{\tau_{beg} - \tau_{end}} \sum_{\tau=\tau_{beg}}^{\tau_{end}} \pi_{t,\tau}^i \quad i = 1, \dots, M$.
    4. $l_t(Y_t = (y_t^1, \dots, y_t^{m_t}) | \tilde{s}_t^n) = \prod_{j=1}^{m_t} [\frac{\pi_t^0}{V} + \sum_{i=1}^M l_t^i(y_t^j; \tilde{s}_t^{n,i}) \hat{\pi}_t^i] \quad n = 1, \dots, N$.
    5. $q_t^n \propto q_{t-1}^n \frac{p(\tilde{s}_t^n) | s_{t-1}^n) \, l_t(Y_t | \tilde{s}_t^n)}{f(\tilde{s}_t | s_{t-1}, y_t)} \quad n = 1, \dots, N$.
  - Return $\widehat{\mathbb{E}} g(X_t) = \sum_{n=1}^N q_t^n g(\tilde{s}_t^n)$.
  - Calculate $\hat{N}_{eff} = \frac{1}{\sum_{n=1}^N (q_t^n)^2}$.
  - Resampling: if $\hat{N}_{eff} < N_{threshold}$: $\begin{cases} s_t^n \sim \sum_{k=1}^N q_t^k \delta_{\tilde{s}_t^k} \\ q_t^n = 1/N \end{cases} \quad n = 1, \dots, N$, else $s_t^n = \tilde{s}_t^n$ for $n = 1, \dots, N$

Figure 2: MTPF: multiple target particle filter with adaptive resampling.

$$\begin{array}{lll} \texttt{Draw } \theta_{\tau+1}^1 & \texttt{from} & p(\theta^1 | Y_{0:t}, \theta_\tau^2, \dots, \theta_\tau^P) \\ & & \vdots \qquad \vdots \\ \texttt{Draw } \theta_{\tau+1}^P & \texttt{from} & p(\theta^P | Y_{0:t}, \theta_{\tau+1}^1, \dots, \theta_{\tau+1}^{P-1}) \end{array}$$

In our case, at a given instant t, the partitionning of θ is:

$$\begin{cases} \theta^j = K_t^j & \text{for } j = 1, \dots, m_t; \\ \theta^{m_t+i} = \pi_t^i & \text{for } i = 1, \dots, M; \\ \theta^{m_t+M+i} = X_t^i & \text{for } i = 1, \dots, M. \end{cases} \tag{5}$$

and the algorithm is described in figure 2. Details the MTPF can be found in [4] and [5].

4 Application to problems with active and passive measurements

We consider a scenario with two targets whose bearings made with an observer are always very closed (see figure 3.2. The trajectories of the targets and of the observer are plotted in figure 3.1. It is to be noted that

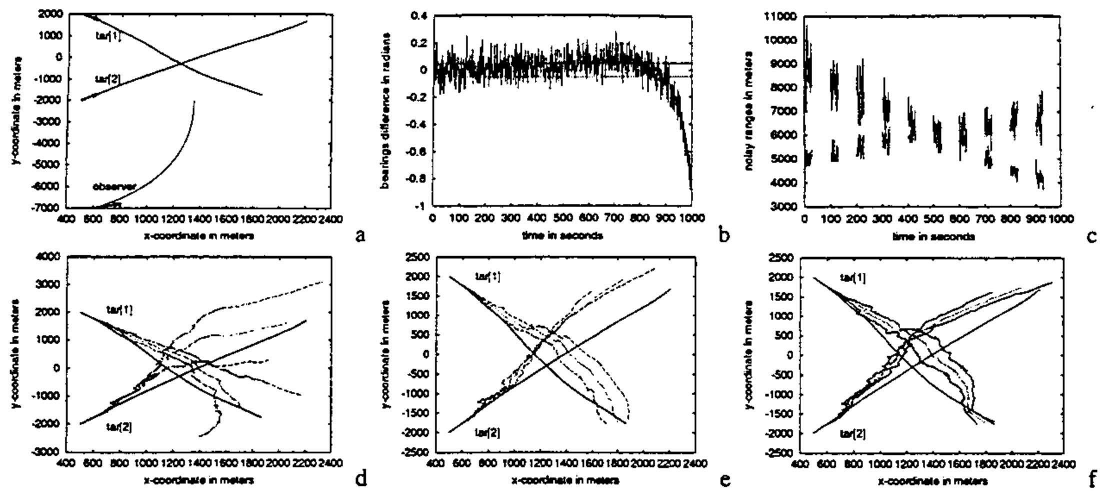

Figure 3: *(a) Trajectories of the targets and of the observer; (b) Difference between the noisy bearings associated with the targets compared with the standard deviation of the measurement noise* $= 0.05$, *i.e.,* 2.8 *degrees; (c) Noisy ranges simulated for* $T = 30$ *and* $P = 100$*; (d-e-f) Averaged estimates (dotted lines) and* 2σ *confidence ellipsoids (dashed lines) obtained with bearings-measurements and* 0%, 20% *and* 50% *of range measurements respectively. The solid lines stand for the real trajectories.*

the observer does not follow a leg by leg trajectory, that renders the estimation of the trajectories quite difficult. As shown in Figures 3.4,5,6, the estimation is ameliorated by adding active measurements (here ranges). We assume that noisy ranges are available during intervals of length T every P times, i.e., if the current time t is such that $t \ mod P \in [0 \ ; T]$. A noisy range associated with the i^{th} target is supposed to follow the equation: $R_t^j = \sqrt{(x_t^i - x_t^{obs})^2 + (y_t^i - y_t^{obs})^2} + Z_t$ where Z_t is a Gaussian noise with standard deviation $\sigma_z|(x_t^i - x_t^{obs})^2 + (y_t^i - y_t^{obs})^2|$, where $\sigma_z = 10^{-5}$. For instance, for $T = 30$ and $P = 100$, the simulated ranges of the two targets are shown in Figure 3.c.

References

[1] J. Diebolt and C. P. Robert. Estimation of finite mixture distributions through Bayesian sampling. *Journal of the Royal Statistical Society series B*, 56:363–375, 1994.

[2] A. Doucet. On sequential simulation-based methods for Bayesian filtering. Technical report, CUED/F-INFENG/TR 310, Signal Processing Group, Departement of Engineering, University of Cambridge, 1998.

[3] N. Gordon, D. Salmond, and A. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proc.F, Radar and signal processing*, 140(2):107–113, April 1993.

[4] C. Hue, J.-P. Le Cadre, and P. Pérez. Tracking multiple objects with particle filtering. Technical report, IRISA, October 2000.

[5] C. Hue, J.-P. Le Cadre, and P. Pérez. The (mr)mtpf: particle filters to track multiple targets using multiple receivers. In *4th International Conference on Information Fusion*, August 2001.

[6] M. Isard and A. Blake. CONDENSATION — Conditional density propagation for visual tracking. *Int. J. Computer Vision*, 29(1):5–28, 1998.

[7] A. Kong, J.S. Liu, and W.H Wong. Sequential imputation method and Bayesian missing data problems. *J. Amer. Statist. Assoc.*, 89:278–288, 1994.

[8] M. Stephens. *Bayesian methods for mixtures of normal distributions*. PhD thesis, Magdalen College, Oxford, 1997.

[9] R. L. Streit and T. E. Luginbuhl. Maximum likelihood method for probabilistic multi-hypothesis tracking. In *Proceedings of SPIE International Symposium, Signal and Data Processing of Small Targets 1994*, volume 2235, Orlando, FL, 5-7 April 1994.