

Sequential Monte Carlo Methods for Multiple Target Tracking and Data Fusion

Carine Hue, Jean-Pierre Le Cadre, *Member, IEEE*, and Patrick Pérez

Abstract—The classical particle filter deals with the estimation of one state process conditioned on a realization of one observation process. We extend it here to the estimation of multiple state processes given realizations of several kinds of observation processes. The new algorithm is used to track with success multiple targets in a bearings-only context, whereas a JPDAF diverges. Making use of the ability of the particle filter to mix different types of observations, we then investigate how to join passive and active measurements for improved tracking.

Index Terms—Bayesian estimation, bearings-only tracking, Gibbs sampler, multiple receivers, multiple targets tracking, particle filter.

I. 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 to be 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.

The main difficulty, however, comes from the assignment of a given measurement to a target model. These assignments are generally unknown, as are the true target models. This is a neat departure from classical estimation problems. Thus, two distinct problems have to be solved *jointly*: the data association and the estimation.

As long as the association is considered in a deterministic way, the possible associations must be exhaustively enumerated. This leads to an NP-hard problem because the number of possible associations increases exponentially with time, as in the multiple hypothesis tracker (MHT) algorithm [28]. In the joint probabilistic data association filter (JPDAF) [11], the association variables are considered to be stochastic variables, and one needs only to evaluate the association probabilities at each time step. However, the dependence assumption on the associations implies the exhaustive enumeration of all possible associations

at the current time step. When the association variables are instead supposed to be statistically independent like in the probabilistic MHT (PMHT [12], [32]), the complexity is reduced. Unfortunately, the above algorithms do not cope with nonlinear models and non-Gaussian noises.

Under such assumptions (stochastic state equation and nonlinear state or measurement equation non-Gaussian noises), particle filters are particularly appropriate. They mainly consist of propagating a weighted set of particles that approximates the probability density of the state conditioned on the observations. Particle filtering can be applied under very general hypotheses, is able to cope with heavy clutter, and is very easy to implement. Such filters have been used in very different areas for Bayesian filtering under different names: The bootstrap filter for target tracking in [15] and the Condensation algorithm in computer vision [20] are two examples, among others. In the earliest studies, the algorithm was only composed of two periods: The particles were predicted according to the state equation during the prediction step; then, their weights were calculated with the likelihood of the new observation combined with the former weights. A resampling step has rapidly been added to dismiss the particles with lower weights and avoid the degeneracy of the particle set into a unique particle of high weight [15]. Many ways have been developed to accomplish this resampling, whose final goal is to enforce particles in areas of high likelihood. The frequency of this resampling has also been studied. In addition, the use of kernel filters [19] has been introduced to regularize the sum of Dirac densities associated with the particles when the dynamic noise of the state equation was too low [26]. Despite this long history of studies, in which the ability of particle filter to track multiple posterior modes is claimed, the extension of the particle filter to multiple target tracking has progressively received attention only in the five last years. Such extensions were first claimed to be theoretically feasible in [2] and [14], but the examples chosen only dealt with one single target. In computer vision, a probabilistic exclusion principle has been developed in [24] to track multiple objects, but the algorithm is very dependent of the observation model and is only applied for two objects. In the same context, a Bayesian multiple-blob tracker (BraMBLe) [6] has just been proposed. It deals with a varying number of objects that are depth-ordered thanks to a 3-D state space. Lately, in mobile robotic [29], a set of particle filters for each target connected by a statistical data association has been proposed. 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.

This work is organized as follows. In Section II, we recall the principles of the basic particle filter with adaptive resampling for a single target. We begin Section III with a presentation of the multitarget tracking problem and its classical solutions

Manuscript received January 31, 2001; revised October 11, 2001. The associate editor coordinating the review of this paper and approving it for publication was Dr. Petar M. Djurić.

C. Hue is with Irisa/Université de Rennes 1, Rennes, France (e-mail: chue@irisa.fr).

J.-P. Le Cadre is with Irisa/CNRS, Rennes, France (e-mail: lecadre@irisa.fr).

P. Pérez is with Microsoft Research, Cambridge, U.K. (e-mail: pperez@microsoft.com).

Publisher Item Identifier S 1053-587X(02)00571-8.

and of the related work on multitarget tracking by particle filtering methods. Then, we present our multitarget particle filter (MTPF). The new algorithm combines the two major steps (prediction and weighting) of the classical particle filter with a Gibbs sampler-based estimation of the assignment probabilities. Then, we propose to add two statistical tests to decide if a target has appeared or disappeared from the surveillance area. Carrying on with the approach of the MTPF, we finally present an extension to multireceiver data in the context of multiple targets (the MRMTPF). Section IV is devoted to experiments. Simulation results in a bearings-only context with a variable clutter density validate the MTPF algorithm. A comparison with a JPDAF based on the extended Kalman filter establishes the superiority of the MTPF for the same scenario. Then, a simulation with occlusion shows how the disappearance of a target can be handled. The suitable quantity and distribution of active measurements are then studied in a particular scenario to improve the performance obtained with passive measurements only.

As far as the notational conventions are concerned, we always use the index i to refer to one among the M tracked targets. The index j designates one of the m_t observations obtained at instant t . The index n is devoted to the N particles. The index τ is used for indexing the iterations in the Gibbs sampler, and r is used for the different receivers. Finally, the probability densities are denoted by p if they are continuous and by \mathbb{P} if they are discrete.

II. BASIC SINGLE-TARGET PARTICLE FILTER

For the sake of completeness, the basic particle filter is now briefly reviewed. 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:

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

It is observed at discrete times via realizations of the stochastic process $(Y_t) \in \mathbb{R}^{n_y}$ governed by the measurement model

$$Y_t = H_t(X_t, W_t). \quad (2)$$

The two processes $(V_t) \in \mathbb{R}^{n_v}$ and $(W_t) \in \mathbb{R}^{n_w}$ in (1) and (2) are only supposed to be independent white noises. Note that the functions F_t and H_t are not assumed linear. We will denote by $Y_{0:t}$ the sequence of the random variables (Y_0, \dots, Y_t) and by $y_{0:t}$ one realization of this sequence. Note that throughout the paper, the first subscript of any vector will always refer to the time.

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., $L_t = p(X_t | Y_0 = y_0, \dots, Y_t = y_t)$ and of estimating any functional $g(X_t)$ of the state by the expectation $\mathbb{E}(g(X_t) | Y_{0:t})$ as well. The recursive Bayesian filter, which is also called the optimal filter, resolves exactly this problem in two steps at each time t .

Suppose we know L_{t-1} . The *prediction step* is done according to the following equation:

$$\begin{aligned} p(X_t = x_t | Y_{0:t-1} = y_{0:t-1}) \\ = \int_{\mathbb{R}^{n_x}} p(X_t = x_t | X_{t-1} = x) L_{t-1}(x) dx. \end{aligned} \quad (3)$$

The observation y_t enables us to correct this prediction using Bayes's rule:

$$L_t(x_t) = \frac{p(Y_t = y_t | X_t = x_t) p(X_t = x_t | Y_{0:t-1} = y_{0:t-1})}{\int_{\mathbb{R}^{n_x}} p(Y_t = y_t | X_t = x) p(X_t = x | Y_{0:t-1} = y_{0:t-1}) dx}. \quad (4)$$

Under the specific assumptions of Gaussian noises V_t and W_t and linear functions F_t and H_t , these equations lead to the Kalman filter's equations. Unfortunately, this modeling is not appropriate in many problems in signal and image processing, which makes the calculation of the integrals in (3) and (4) infeasible (no closed-form).

The original particle filter, which is called the bootstrap filter [15], proposes to approximate the densities L_t by a finite weighted sum of N Dirac densities centered on elements of \mathbb{R}^{n_x} , which are called particles.

The application of the bootstrap filter requires that one knows how to do the following:

- sample from initial prior marginal $p(X_0)$;
- sample from $p(V_t)$ for all t ;
- compute $p(Y_t = y_t | X_t = x_t)$ for all t through a known function l_t such that $l_t(y; x) \propto p(Y_t = y | X_t = x)$, where missing normalization must not depend on x .

The algorithm then evolves the particle set $S_t = (s_t^n, q_t^n)_{n=1, \dots, N}$, 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 $L_{S_t} = \sum_{n=1}^N q_t^n \delta_{s_t^n}$. In the bootstrap filter, the particles are "moved" by sampling from the dynamics (1), and importance sampling theory shows that the weighting is only based on likelihood evaluations. In the most general setting [9], the displacement of particles is obtained by sampling from an appropriate density f , which might depend on the data as well. The complete procedure is summarized in Fig. 1. Some convergence results of the empirical distributions to the posterior distribution on the path space have been proved when the number N of particles tends toward infinity [25], [10]. In the path space $(\mathbb{R}^{n_x})^{t+1}$, each particle s_t^n at time t can be considered to be a discrete path of length $t + 1$. Compared with the particle filter presented in Fig. 1, particle filtering in the space of paths consists of incrementing the particle state space at each time step and representing each particle by the concatenation of the new position at time t and the set of previous positions between times 0 and $t - 1$. In [25], the fluctuations on path space of the so-called interacting particle systems are studied. In the context of sequential Monte Carlo methods [10] that cover most of the particle filtering methods proposed in the last few years, the convergence and the rate of convergence of order $1/N$ of the average mean square error is proved. Under more restrictive hypotheses, the almost-sure convergence is proved as well [10].

To evaluate the degeneracy of the particle set, the effective sample size has been defined [21], [23]. As advocated in [9], a resampling step is performed in the algorithm presented in Fig. 1 in an adaptive way when the effective sample size, estimated by \hat{N}_{eff} , is under a given threshold. It avoids to obtain a degenerate

- 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: $\begin{cases} \text{Compute un-normalized weights: } \tilde{q}_t^n = q_{t-1}^n \frac{p(\tilde{s}_t^n | s_{t-1}^n) l_t(y_t; \tilde{s}_t^n)}{f(\tilde{s}_t^n | s_{t-1}^n, y_t)} \text{ for } n = 1, \dots, N. \\ \text{Normalize weights: } q_t^n = \frac{\tilde{q}_t^n}{\sum_{n=1}^N \tilde{q}_t^n} \text{ for } n = 1, \dots, N. \end{cases}$
 - Return $\hat{\mathbb{E}}g(X_t) = \sum_{n=1}^N q_t^n g(\tilde{s}_t^n).$
 - Calculate $\hat{N}_{eff} = \frac{1}{\sum_{n=1}^N (q_t^n)^2}.$
 - Resampling: if $\hat{N}_{eff} < N_{threshold}$: $\begin{cases} s_t^n \sim \sum_{k=1}^N q_t^k \delta_{\tilde{s}_t^k} \\ q_t^n = 1/N \end{cases} \quad n = 1, \dots, N, \text{ else } s_t^n = \tilde{s}_t^n \text{ for } n = 1, \dots, N.$

Fig. 1. Basic particle filter with adaptive resampling.

particle set composed of only few particles with high weights and all the others with very small ones.

Beside the discretization of the filtering integrals, the use of such particles enables the maintenance of multiple hypotheses on the position of the target and to keep in the long term only the particles whose position is likely given the whole sequence of observations.

We find more details on the algorithm in [9] or [15] and on adaptive resampling in [9] and [21]. After these recalls, let us present briefly the multitarget tracking problem and its classical solutions, as well as the existing works on particle filtering methods for MTT. Then, we will propose the MTPF.

III. MULTITARGET PARTICLE FILTER

A. MTT Problem and Its Classical Treatment

Let M be the number of targets to track that are assumed to be known and fixed for the moment (the case of a varying unknown number will be addressed in Section III-C). The index i designates one among the M targets and is always used as first superscript. Multitarget tracking consists of estimating the state vector made by concatenating the state vectors of all targets. It is generally assumed that the targets are moving according to independent Markovian dynamics. At time t , $X_t = (X_t^1, \dots, X_t^M)$ follows the state equation (1) decomposed in M partial equations

$$X_t^i = F_t^i(X_{t-1}^i, V_t^i) \quad \forall i = 1, \dots, M. \quad (5)$$

The noises (V_t^i) and $(V_t^{i'})$ are supposed only to be white both temporally and spatially and independent for $i \neq i'$.

The observation vector collected at time t is denoted by $y_t = (y_t^1, \dots, y_t^{m_t})$. The index j is used as first superscript to refer to one of the m_t measurements. The vector y_t is composed of detection measurements and clutter measurements. The false alarms are assumed 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 is the number of false alarms per unit volume. As we do not know the origin of each measurement, one has to introduce the vector K_t to describe the associations between the measurements and the targets. Each component K_t^j is a random variable that takes its values among $\{0, \dots, M\}$. Thus, $K_t^j = i$

indicates that y_t^j is associated with the i th target. In this case, y_t^j is a realization of the stochastic process

$$Y_t^j = H_t^i(X_t^i, W_t^j) \quad \text{if } K_t^j = i. \quad (6)$$

Again, the noises (W_t^j) and $(W_t^{j'})$ are supposed only to be white noises, independent for $j \neq j'$. We assume that the functions H_t^i are such that they can be associated with functional forms l_t^i such that

$$l_t^i(y; x) \propto p(Y_t^j = y | K_t^j = i, X_t^i = x).$$

We dedicate the model 0 to false alarms. Thus, if $K_t^j = 0$, the j th measurement is associated with the clutter, but we do not associate any kinematic model to false alarms.

As the indexing of the measurements is arbitrary, all the measurements have the same *a priori* probability to be associated with a given model i . At time t , these association probabilities define the vector $\pi_t = (\pi_t^0, \pi_t^1, \dots, \pi_t^M) \in [0, 1]^{M+1}$. Thus, for $i = 1, \dots, M$, $\pi_t^i \triangleq \mathbb{P}(K_t^j = i)$ for all $j = 1, \dots, m_t$ is the discrete probability that any measurement is associated with the i th target.

To solve the data association, some assumptions are commonly made [3].

- A1) One measurement can originate from one target or from the clutter.
- A2) One target can produce zero or one measurement at one time.

The assumption A1) expresses that the association is exclusive and exhaustive. Consequently, $\sum_{i=0}^M \pi_t^i = 1$.

Assumption A2) implies that m_t may differ from M and, above all, that the association variables K_t^j for $j = 1, \dots, m_t$ are dependent.

Under these assumptions, the MHT algorithm [28] builds recursively the association hypotheses. One advantage of this algorithm is that the appearance of a new target is hypothesized at each time step. However, the complexity of the algorithm increases exponentially with time. Some pruning solutions must be found to eliminate some of the associations.

The JPDAF begins with a gating of the measurements. Only the measurements that are inside an ellipsoid around the predicted state are kept. The gating assumes that the measurements are distributed according to a Gaussian law centered on the predicted state. Then, the probabilities of each association $K_t^j = i$ are estimated. As the variables K_t^j are assumed dependent by

A2), this computation implies the exhaustive enumeration of all the possible associations K_t^l for $l \neq j$.

The novelty in the PMHT algorithm [12], [32], [33] consists of replacing the assumption A2) by A3):

- A3) One target can produce zero or several measurements at one time.

This assumption is often criticized because it does not match the physical reality. However, from a mathematical point of view, it ensures the stochastic independence of the variables K_t^j and it drastically reduces the complexity of the π_t vector estimation.

The assumptions A1) and A3) will be kept in the MTPF presented later. Let us present now the existing works solving MTT with particle filtering methods.

B. Related Work: MTT With Particle Filtering Methods

In the context of multitarget tracking, particle filtering methods are appealing: As the association needs only to be considered at a given time iteration, the complexity of data association is reduced. First, two extensions of the bootstrap filter have been considered. In [2], a bootstrap-type algorithm is proposed in which the sample state space is a “(multitarget) state space.” However, nothing is said about the association problem that needs to be solved to evaluate the sample weights. It is, in fact, the ability of the particle filtering to deal with multimodality due to (high) clutter that is pointed out compared with deterministic algorithms like the nearest neighbor filter or the probabilistic data association (PDA) filter. No examples with multiple targets are presented. The simulations only deal with a single target in clutter with a linear observation model. In [14], a hybrid bootstrap filter is presented where the particles evolve in a single-object state space. Each particle gives a hypothesis on the state of one object. Thus, the *a posteriori* law of the targets, given the measurements, is represented by a Gaussian mixture. Each mode of this law then corresponds to one of the objects. However, as pointed out in [14], the likelihood evaluation is possible only under the availability of the “prior probabilities of all possible associations between” the measurements and the targets. It may be why the simulation example only deals with one single target in clutter. Even if the likelihood could be evaluated, the way to represent the *a posteriori* law by a mixture can lead to the loss of one of the targets during occlusions. The particles tracking an occluded target get very small weights and are therefore discarded during the resampling step. This fact has been pointed out in [29].

In image analysis, the Condensation algorithm has been extended to the case of multiple objects as well. In [24], the case of two objects is considered. The hidden state is the concatenation of the two single-object states and of a binary variable indicating which object is closer to the camera. This latter variable solves the association during occlusion because the measurements are affected to the foreground object. Moreover, a probabilistic exclusion principle is integrated to the likelihood measurement to penalize the hypotheses with the two objects overlapping. In [6], the state is composed of an integer equal to the number of objects and of a concatenation of the individual states. A three-dimensional (3-D) representation of the objects

gives access to their depth ordering, thus solving the association issue during occlusions. Finally, in mobile robotics [29], a particle filter is used for each object tracked. The likelihood of the measurements is written like in a JPDAF. Thus, the assignment probabilities are evaluated according to the probabilities of each possible association. Given these assignment probabilities, the particle weights can be evaluated. The particle filters are then dependent through the evaluation of the assignment probabilities. Independently of the two latter works [6] and [29], we have developed the MTPF, where the data association is approached in the same probabilistic spirit as the basic PMHT [12], [32].

First, to estimate the density $L_t = p(X_t = (X_t^1, \dots, X_t^M) | Y_{0:t} = y_{0:t})$ with particle filtering methods, we must choose the state space for the particles. As mentioned before, a unique particle filter with a single-target state space seemed to us a poor choice as the particles tracking an occluded object would be quickly discarded. We have considered using one particle filter per object but without finding a consistent way to make them dependent. The stochastic association vector K_t introduced in Section III-A could also be considered to be an additional particle component. However, as the ordering of the measurements is arbitrary, it would not be possible to devise a dynamic prior on it. Moreover, the state space would increase, further making the particle filter less effective. Finally, we have chosen to use particles whose dimension is the sum of those of the individual state spaces corresponding to each target, as in [6] and [24]. Each of these concatenated vectors then gives jointly a representation of all targets. Let us describe the MTPF. Further details on the motivations for the different ingredients of the MTPF can be found in [18].

C. MTPF Algorithm

Before describing the algorithm itself, let us first notice that the association probability π_t^0 that a measurement is associated with the clutter is a constant that can be computed

$$\pi_t^0 = \mathbb{P}(K_t^j = 0) \quad (7)$$

$$= \sum_{l=0}^{m_t} \mathbb{P}(K_t^j = 0 | N_t^0 = l) \mathbb{P}(N_t^0 = l) \quad (8)$$

$$= \sum_{l=0}^{m_t} \frac{l}{m_t} \exp(-\lambda V) \frac{(\lambda V)^l}{l!} \quad (9)$$

where N_t^0 is the number of measurements arising from the clutter at time t . Assuming that there are l clutter originated measurements among the m_t measurements collected at time t , the *a priori* probability that any measurement comes from the clutter is equal to l/m_t ; hence, we get the equality $\mathbb{P}(K_t^j = 0 | N_t^0 = l) = l/m_t$ used to derive (9) from (8).

The initial set $S_0 = (s_0^n, 1/N)_{n=1, \dots, N}$ of N particles is such that each component $s_0^{n,i}$ for $i = 1, \dots, M$ is sampled from $p(X_0^i)$ independently from the others. Assume we have obtained $S_{t-1} = (s_{t-1}^n, q_{t-1}^n)_{n=1, \dots, N}$ with $\sum_{n=1}^N q_{t-1}^n = 1$. Each particle is a vector of dimension $\sum_{i=1}^M n_x^i$, where we denote by $s_{t-1}^{n,i}$ the i th component of s_{t-1}^n and where n_x^i designates the dimension of target i .

The prediction is performed by sampling from some proposal density f . In the bootstrap filter case, f coincides with the dynamics (5)

$$\tilde{s}_t^n = \begin{pmatrix} F_t^1(s_{t-1}^{n,1}, v_t^{n,1}) \\ \vdots \\ F_t^M(s_{t-1}^{n,M}, v_t^{n,M}) \end{pmatrix}, \quad n = 1, \dots, N. \quad (10)$$

Examine now the computation of the likelihood of the observations conditioned on the n th particle. We can write for all $n = 1, \dots, N$

$$\begin{aligned} p(Y_t = (y_t^1, \dots, y_t^{m_t}) | X_t = \tilde{s}_t^n) \\ = \prod_{j=1}^{m_t} p(y_t^j | \tilde{s}_t^n) \propto \prod_{j=1}^{m_t} \left[\frac{\pi_t^0}{V} + \sum_{i=1}^M l_t^i(y_t^j; \tilde{s}_t^{n,i}) \pi_t^i \right]. \end{aligned} \quad (11)$$

Note that the first equality in (11) is true only under the assumption of conditional independence of the measurements, which we will make. To derive the second equality in (11), we have used the total probability theorem with the events $(K_t^j = i)_{i=0, \dots, M}$ and under the supplementary assumption that the normalization factors between l_t^i and $p(Y_t^j = y | K_t^j = i, X_t^j = x)$ is the same for all i .

We still need to estimate at each time step the association probabilities $(\pi_t^i)_{i=1, \dots, M}$, which can be seen as the stochastic coefficients of the M -component mixture. Two main ways have been found in the literature to estimate the parameters of this model: the expectation maximization (EM) method (and its stochastic version the SEM algorithm [5]) and the data augmentation method. The second one amounts in fact to a Gibbs sampler.

In [12], [32], and [33], the EM algorithm is extended and applied to multitarget tracking. This method implies that the vectors π_t and X_t are considered as parameters to estimate. The maximization step can be easily conducted in the case of deterministic trajectories, and the additional use of a maximum *a posteriori* (MAP) estimate enables the achievement of it for nondeterministic trajectories. Yet, the nonlinearity of the state and observation functions makes this step very difficult. Finally, the estimation is done iteratively in a batch approach we would like to avoid. For these reasons, we have not chosen an EM algorithm to estimate the association probabilities.

The data augmentation algorithm is quite different in its principle. The vectors X_t , K_t , and π_t are considered to be random variables with prior densities. Samples are then obtained iteratively from their joint posterior using a proper Markov chain Monte Carlo (MCMC) technique, namely, the Gibbs sampler. This method has been studied in [4], [8], [13], [30], and [31], for instance. It can be run sequentially at each time period. The Gibbs sampler is a special case of the Metropolis–Hasting algorithm with the proposal densities being the conditional distributions and the acceptance probability being consequently always equal to one. See [7] for an introduction to MCMC simulation methods and also for a presentation of the EM algorithm.

Let Π_t be the stochastic variable associated with the probability π_t . 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 get a partition $\theta_t^1, \dots, \theta_t^P$ of θ_t and to sample alternatively

from the conditional posterior distribution of each component of the partition. Let the index τ denote the iterations in the Gibbs sampler. The second subscript of the vectors refers to the iteration counter. Assume that the $\tau + 1$ first elements of the Markov chain $(\theta_{t,0}, \dots, \theta_{t,\tau})$ have been drawn. We sample the P components of $\theta_{t,\tau+1}$ as follows:

$$\begin{aligned} \text{Draw } \theta_{t,\tau+1}^1 & \text{ from } p(\theta_t^1 | Y_{0:t}, \theta_{t,\tau}^2, \dots, \theta_{t,\tau}^P) \\ \text{Draw } \theta_{t,\tau+1}^2 & \text{ from } p(\theta_t^2 | Y_{0:t}, \theta_{t,\tau+1}^1, \theta_{t,\tau}^3, \dots, \theta_{t,\tau}^P) \\ & \vdots \\ \text{Draw } \theta_{t,\tau+1}^P & \text{ from } p(\theta_t^P | Y_{0:t}, \theta_{t,\tau+1}^1, \dots, \theta_{t,\tau}^{P-1}). \end{aligned}$$

In our case, at a given instant t , we follow this algorithm with

$$\begin{cases} \theta_t^j = K_t^j, & \text{for } j = 1, \dots, m_t \\ \theta_t^{m_i+i} = \Pi_t^i, & \text{for } i = 1, \dots, M \\ \theta_t^{m_t+M+i} = X_t^i, & \text{for } i = 1, \dots, M. \end{cases} \quad (12)$$

The initialization of the Gibbs sampler consists of assigning uniform association probabilities, i.e., $\pi_{t,0}^i = (1 - \pi_t^0)/M$ for all $i = 1, \dots, M$, and taking $X_{t,0} = \sum_{n=1}^N q_{t-1}^n \tilde{s}_t^n$, i.e., the centroid of the predicted particle set. The K_t variables do not need initializing because at the first time step of the Gibbs sampler, they will be sampled conditioned on $\pi_{t,0}^i$, $i = 1, \dots, M$ and $X_{t,0}$. Then, suppose that at instant t , we have already simulated $(\theta_{t,1}, \dots, \theta_{t,\tau})$. The $\tau + 1$ th iteration is handled as follows.

- As the $(K_t^j)_{j=1, \dots, m_t}$ are assumed to be independent, their individual discrete conditional density reads

$$\mathbb{P}(K_t^j | Y_{0:t}, X_t, (K_t^l)_{l \neq j}, \Pi_t) = \mathbb{P}(K_t^j | Y_t^j, X_t, \Pi_t). \quad (13)$$

Assignment variables K_t^j are discrete, and we can write¹

$$\begin{aligned} \mathbb{P}(K_t^j = i | Y_t^j = y_t^j, X_t, \Pi_t) \\ = \frac{p(Y_t^j = y_t^j | K_t^j = i, X_t, \Pi_t) \mathbb{P}(K_t^j = i | X_t, \Pi_t)}{p(Y_t^j = y_t^j | X_t, \Pi_t)} \\ \propto \begin{cases} \pi_t^i l_t^i(y_t^j; x_t^i), & \text{if } i = 1, \dots, M \\ \pi_t^0/V, & \text{if } i = 0. \end{cases} \end{aligned} \quad (14)$$

The realizations $k_{t,\tau+1}^j$ of the vector $K_{t,\tau+1}$ are then sampled according to the weights $p_{t,\tau+1}^{j,0} = \pi_t^0/V$

$$p_{t,\tau+1}^{j,i} = \pi_{t,\tau}^i l_t^i(y_t^j; x_{t,\tau}^i) \quad \text{for } i = 1, \dots, M.$$

- Mixture proportion vector $\Pi_{t,\tau+1}^{1:M}$ is drawn from the conditional density

$$\begin{aligned} p(\Pi_{t,\tau+1}^{1:M} | K_{t,\tau+1}, X_{t,\tau}, Y_{0:t}) \\ = p(\Pi_{t,\tau+1}^1, \dots, \Pi_{t,\tau+1}^M | K_{t,\tau+1}^1, \dots, K_{t,\tau+1}^M, X_{t,\tau}, Y_{0:t}) \\ = p(\Pi_{t,\tau+1}^1, \dots, \Pi_{t,\tau+1}^M | K_{t,\tau+1}^1, \dots, K_{t,\tau+1}^M) \\ \propto \text{Dirichlet}(\Pi_{t,\tau+1}^{1:M} | \{1 + n^i(K_{t,\tau+1})\}_{i=1, \dots, M}) \end{aligned} \quad (15)$$

where we denote by $n^i(K)$ the number of k^j equal to i and where $\text{Dirichlet}(\delta_1, \dots, \delta_M)$ denotes the Dirichlet distribution on the simplex $\{(\pi_1^1, \dots, \pi_1^{M-1}, 1 - \pi_1^1 -$

¹Using Bayes's rule, $p(a|b, c) = (p(b|a, c)p(a|c))/p(b|c)$.

$\dots - \pi_t^{M-1}$): $\pi_t^1 + \dots + \pi_t^{M-1} \leq 1$ with density proportional to $\pi_t^{1(\delta_{t1}-1)} \times \dots \times \pi_t^{M-1(\delta_{M-1}-1)} \times (1 - \pi_t^1 - \dots - \pi_t^{M-1})^{\delta_M-1}$. The vector $\Pi_{t,\tau+1}^{1:M}$ is first drawn according to (15) and then normalized to restore the sum $\sum_{i=1}^M \pi_{t,\tau+1}^i$ to $1 - \pi_t^0$.

- $X_{t,\tau+1}$ has to be sampled according to the density

$$p(X_t|Y_{0:t}, K_{t,\tau+1}, \Pi_{t,\tau+1}) = \prod_{i=1}^M p(X_t^i|Y_{0:t}, K_{t,\tau+1}, \Pi_{t,\tau+1}). \quad (16)$$

The values of $K_{t,\tau+1}$ might imply that one target is associated with zero or several measurements. Let us define $\mathcal{M}_{t,\tau+1}^i = \{j \in [1 \dots m_t]: K_{t,\tau+1}^j = i\}$. Hence, we decompose the preceding product in two products:

$$\prod_{i: \mathcal{M}_{t,\tau+1}^i \neq \emptyset} p(X_t^i|Y_{0:t-1}, y_t^{\mathcal{M}_{t,\tau+1}^i}, \Pi_{t,\tau+1}) \prod_{i: \mathcal{M}_{t,\tau+1}^i = \emptyset} p(X_t^i|Y_{0:t-1}, \Pi_{t,\tau+1}) \quad (17)$$

where $y_t^{\mathcal{M}_{t,\tau+1}^i} = \{y_t^j, j \in \mathcal{M}_{t,\tau+1}^i\}$. The first product contains the targets that are associated with at least one measurement under the association $K_{t,\tau+1}$. In this case, the measurements are denoted by $y_t^{\mathcal{M}_{t,\tau+1}^i}$. The second product contains the targets that are associated with no measurements under $K_{t,\tau+1}$.

- Let i be an index in the first product. We can write

$$p(X_t^i|Y_{0:t-1}, y_t^{\mathcal{M}_{t,\tau+1}^i}, \Pi_{t,\tau+1}) = \frac{p(y_t^{\mathcal{M}_{t,\tau+1}^i}|X_t^i) p(X_t^i|Y_{0:t-1})}{p(y_t^{\mathcal{M}_{t,\tau+1}^i}|Y_{0:t-1})}. \quad (18)$$

We are not able to sample directly from the density $(p(y_t^{\mathcal{M}_{t,\tau+1}^i}|X_t^i) p(X_t^i|Y_{0:t-1})) / (p(y_t^{\mathcal{M}_{t,\tau+1}^i}|Y_{0:t-1}))$ for the same reasons as those exposed in Section II to justify the use of the particle filter (intractability of the integrals). We propose to build the particle set $\Sigma_{\tau+1} = (\sigma_{\tau+1}^n, \chi_{\tau+1}^n)_{n=1, \dots, N}$, whose weights $\chi_{\tau+1}^n$ measure the likelihood of the observations associated by $K_{t,\tau+1}$ to target X_t^i . More precisely, we let

$$\begin{cases} \sigma_{\tau+1}^n = \hat{s}_t^{n,i}; \\ \chi_{\tau+1}^n = \frac{p(y_t^{\mathcal{M}_{t,\tau+1}^i}|X_t^i = \sigma_{\tau+1}^n) q_{t-1}^n}{\sum_{k=1}^N p(y_t^{\mathcal{M}_{t,\tau+1}^i}|X_t^i = \sigma_{\tau+1}^k) q_{t-1}^k}. \end{cases} \quad (19)$$

As the predicted empirical law $L_{\hat{s}_t} = \sum_{n=1}^N q_{t-1}^n \delta_{\sigma_{\tau+1}^n}^i$ is “close” to the predicted law $p(X_t^i|Y_{0:t-1})$, we expect the empirical distribution $\Lambda_{\tau+1} = \sum_{n=1}^N \chi_{\tau+1}^n \delta_{\sigma_{\tau+1}^n}^i$ to be close to $p(X_t^i|y_t^{\mathcal{M}_{t,\tau+1}^i}, Y_{0:t-1})$. However, the weak convergence of $\Lambda_{\tau+1}$ to

$p(X_t^i|y_t^{\mathcal{M}_{t,\tau+1}^i}, Y_{0:t-1})$ when N tends toward infinity remains to be proved. Not being able to sample from $p(X_t^i|y_t^{\mathcal{M}_{t,\tau+1}^i}, Y_{0:t-1})$, we draw $X_{t,\tau+1}^i$ as a realization from $\Lambda_{\tau+1}$.

- Now, let i be an index in the second product. As we do not have any measurement to correct the predicted particles, we draw a realization from the density $\sum_{n=1}^N q_{t-1}^n \delta_{s_t^n}^i$ for $X_{t,\tau+1}^i$.

After a finite number of iterations, we estimate the vector π_t by the average of its last realizations:

$$\hat{\pi}_t^i = \frac{1}{\tau_{beg} - \tau_{end}} \sum_{\tau=\tau_{beg}}^{\tau_{end}} \pi_{t,\tau}^i. \quad (20)$$

Finally, the weights can be computed according to (11) using the estimate $\hat{\pi}_t^i$ of π_t^i . By construction, $\hat{\pi}_t^i$ follows the law $p(\Pi_t^i|Y_{0:t})$. Thus, the use of a Gibbs sampler enables us to take into account the current measurements. Consequently, the estimates measure in a way the *a posteriori* detection probability of each target. It improves the estimation of the targets because the measurements contribute to the estimation proportionally to these probabilities $\hat{\pi}_t$. Moreover, the *a priori* probability of detecting a target, which is usually denoted by P_d , is not needed in the MTPF. This probability is needed when the associations are considered in classical algorithms like the PMHT or the JPDAF.

The resampling step is performed in an adaptive way when the estimated effective sample size \hat{N}_{eff} is under the threshold $N_{threshold}$. In [18], we have compared the performance of particle filtering with systematic and adaptive resampling in a single-object bearings-only synthetic example. The bias on the estimation error was almost the same, but the standard deviation was twice as small using adaptive resampling rather than systematic resampling.

Due to the estimation of the π_t vector needed for the computation of the particles likelihood, the convergence of the MTPF could be affected. It could be interesting to evaluate the error on the estimate of X_t implied by the error made on the estimate of π_t . This is not addressed in this work.

The complete procedure is summarized in Fig. 2.

D. Varying Number of Targets

Until now, the number of targets M was assumed known and constant. We now make suggestions to relax this constraint. As per the disappearance of objects, the vector π_t provides useful information: the disappearance of one target from the surveillance area can be detected by a drop in the corresponding π_t component. We will use the estimate of π_t to decide between the two following hypotheses.

- (H_0^D) —The target is present in the surveillance area.
- (H_1^D) —The target is not present in the surveillance area.

If the target is still present in the surveillance area, the fall of π_t can only be due to its nondetection, which occurs with a probability $1 - P_d$. Let D_t^i be the binary variable equal to 1 if the i th target has been detected at time t and 0 otherwise. Over a test interval $[t_1, \dots, t_d]$ and for a given target i , the variables $D_{t_1}^i, \dots, D_{t_d}^i$ are distributed according to a binomial law of

- 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. $\pi_{t,\tau+1}^{1:M} \sim \text{Dirichlet}((1 + n^i(K_{t,\tau+1}))_{i=1,\dots,M}), n^i(K) \triangleq \#\{j : K^j = i\}.$
 - c. For each i such that $\mathcal{M}_{t,\tau+1}^i \triangleq \{j \in [1 \dots m_t] : K_{t,\tau+1}^j = i\} \neq \emptyset$,
 - ▶ $\begin{cases} \sigma_{\tau+1}^n = \tilde{s}_t^{n,i} \\ \chi_{\tau+1}^n = \frac{p(y_t^{\mathcal{M}_{t,\tau+1}^i} | X_t^i = \sigma_{\tau+1}^n) q_{t-1}^n}{\sum_{n=1}^N p(y_t^{\mathcal{M}_{t,\tau+1}^i} | X_t^i = \sigma_{\tau+1}^n) q_{t-1}^n} \end{cases} \quad n = 1, \dots, N.$
 - ▶ $X_{t,\tau+1}^i \sim \sum_{n=1}^N \chi_{\tau+1}^n \delta_{\sigma_{\tau+1}^n}$
 - d. For each i such that $\mathcal{M}_{t,\tau+1}^i = \emptyset$, $X_{t,\tau+1}^i \sim \sum_{n=1}^N q_{t-1}^n \delta_{\tilde{s}_t^{n,i}}.$
- 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. $q_t^n \propto q_{t-1}^n \frac{p(\tilde{s}_t^n | s_{t-1}^n) l_t(Y_t | \tilde{s}_t^n)}{f(\tilde{s}_t^n | s_{t-1}^n, y_t)} \quad n = 1, \dots, N.$
- Return $\hat{\mathbb{E}}g(X_t) = \sum_{n=1}^N q_t^n g(\tilde{s}_t^n).$
- Calculate $\hat{N}_{eff} = \frac{1}{\sum_{n=1}^N (q_t^n)^2}.$
- Resampling: if $\hat{N}_{eff} < N_{threshold}$: $\begin{cases} s_t^n \sim \sum_{k=1}^N q_t^k \delta_{\tilde{s}_t^k} & n = 1, \dots, N, \\ q_t^n = 1/N \end{cases}$
 else $s_t^n = \tilde{s}_t^n$ for $n = 1, \dots, N$

Fig. 2. MTPF: Particle filter algorithm for multiple targets with adaptive resampling.

parameters $(P_d, 1 - P_d)$. These variables are unknown, but we can use the estimates $\hat{\pi}_t^i$ to estimate them. Let us define

$$\begin{cases} \hat{D}_t^i = 1, & \text{if } \hat{\pi}_t^i > D_{threshold} \\ \hat{D}_t^i = 0, & \text{otherwise} \end{cases} \quad (21)$$

where $D_{threshold}$ is a probability threshold. The χ^2 test with the obtained variables $\hat{D}_{t_1}^i, \dots, \hat{D}_{t_d}^i$ decides on the true hypothesis. This test consists of computing the distance $\alpha^2 = \sum_{classes} ((O_c - E_c)^2) / E_c$ between the expected size E_c and the obtained size O_c of each class (here, the class 0 and the class 1). When t_d tends toward infinity, α^2 is asymptotically distributed as a χ^2 law. One admits that α^2 is reasonably approximated by a χ^2 law under the conditions that the expected size of each class

is higher than 4. That is why in practice, the length of the interval $t_d - t_1$ must be chosen such that $(t_d - t_1)(1 - P_d) > 4$.

As far as the algorithm is concerned, this reduction only leads to update M (the number of targets) and to remove the components of the particles related to the disappeared target. It can be integrated to the MTPF as described in Fig. 3.

On the other hand, the arrival of a new target might be related to an observation whose likelihood is low, whatever target it is associated with. As a result, assignment variables simulated by the Gibbs sampler might be more often equal to 0. We propose to use the values of the assignment variables to decide between the two following hypotheses.

- (H_0^A) —A new target is arriving inside the surveillance area.
- (H_1^A) —No new target is arriving inside the surveillance area.

- 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 – 3. $\hat{\pi}_t^i = \frac{1}{\tau_{beg} - \tau_{end}} \sum_{\tau=\tau_{beg}}^{\tau_{end}} \pi_{t,\tau}^i \quad i = 1, \dots, M$ obtained by Gibbs sampling (see Fig. 2)
 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. $q_t^n \propto q_{t-1}^n \frac{p(\tilde{s}_t^n | s_{t-1}^n) l_t(Y_t | \tilde{s}_t^n)}{f(\tilde{s}_t^n | s_{t-1}^n, y_t)} \quad n = 1, \dots, N.$
 - 6. Disappearance test:
 - △ Calculate $\begin{cases} \hat{D}_t^i = 1 \text{ if } \hat{\pi}_t^i > D_{threshold} \\ \hat{D}_t^i = 0 \text{ otherwise} \end{cases} \quad i = 1, \dots, M$
 - △ For $i = 1, \dots, M$, test the hypothesis (H_0^D) against (H_1^D) with a χ^2 .
 - If (H_1^D) is decided, replace M by $M - 1$ and remove the particle components related to target i .
 - 7. Appearance test:
 - △ Calculate $\hat{N}_t^0 = \frac{1}{\tau_{beg} - \tau_{end}} \sum_{\tau=\tau_{beg}}^{\tau_{end}} n_t^0(K_{t,\tau})$.
 - △ Test the hypothesis (H_0^A) against (H_1^A) with a χ^2 .
 - If (H_1^A) is decided, replace M by $M + 1$ and initialize the new particle components.
- Return $\hat{\mathbb{E}}g(X_t) = \sum_{n=1}^N q_t^n g(\tilde{s}_t^n)$.
- Calculate $\hat{N}_{eff} = \frac{1}{\sum_{n=1}^N (q_t^n)^2}$.
- Resampling: if $\hat{N}_{eff} < N_{threshold}$: $\begin{cases} s_t^n \sim \sum_{k=1}^N q_t^k \delta_{\tilde{s}_t^k} \\ q_t^n = 1/N \end{cases} \quad n = 1, \dots, N.$

Fig. 3. Particle filter algorithm for multiple targets with adaptive resampling and varying target number.

Let \hat{N}_t^0 be the estimate of N_t^0 , which is the number of measurements arising from the clutter at time t , supplied by the Gibbs sampler

$$\hat{N}_t^0 = \frac{1}{\tau_{beg} - \tau_{end}} \sum_{\tau=\tau_{beg}}^{\tau_{end}} n_t^0(K_{t,\tau}) \quad (22)$$

where $n_t^0(K) = \#\{j: K^j = 0\}$. Over an interval $[t_1, \dots, t_d]$, a χ^2 test enables us to measure the adequation between the Poisson law of parameter λV followed by $(N_t^0)_{t=t_1, \dots, t_d}$ and the empirical law of the variables $(\hat{N}_t^0)_{t=t_1, \dots, t_d}$. This test can also be integrated to the MTPF, as described in Fig. 3. Nevertheless, the initialization of the new target based on the observation sets is a tricky problem, which we have not solved yet.

E. Multireceiver Multitarget Particle Filter—MRMTPF

A natural extension of the MTPF is to consider that observations can be collected by multiple receivers. Let R be their number. We will see that we can easily adapt the particle filter to this situation. We always consider that the M targets (their number is fixed again) obey the state equation (5). Some

useful notations must be added to modify the measurement equations. The observation vector at time t will be denoted by $y_t = (y_{t,r^1}^1, \dots, y_{t,r^{m_t}}^{m_t})$, where r^j refers to the receiver that received the j th measure. This measurement is then a realization of the stochastic process

$$Y_{t,r^j}^j = H_{t,r^j}^i(X_t^i, W_t^j), \quad \text{if } K_t^j = i. \quad (23)$$

We assume the independence of the observations collected by the different receivers. We denote by $l_{t,r^j}^i(y; x)$ the functions that are proportional to $p(Y_{t,r^j}^j = y | K_t^j = i, X_t^i = x)$. The likelihood of the observations conditioned by the n th particle is readily obtained as

$$\begin{aligned} p(Y_t = (y_{t,r^1}^1, \dots, y_{t,r^{m_t}}^{m_t}) | X_t = s_t^n) \\ = \prod_{j=1}^{m_t} p(y_{t,r^j}^j | s_t^n) \\ \propto \prod_{j=1}^{m_t} \left[\frac{\pi_t^0}{V} + \sum_{i=1}^M l_{t,r^j}^i(y_{t,r^j}^j; s_t^{n,i}) \hat{\pi}_t^i \right]. \end{aligned} \quad (24)$$

There is no strong limitation on the use of the particle filter for multireceiver and multitarget tracking: the MRMPF is obtained from the MTPF by replacing the likelihood functions $l_t^i(y; x)$ by the functions $l_{t,r}^i(y; x)$.

Moreover, it can deal with measurements of varied periodicities. We present in Section IV-B a scenario mixing active and passive measurements.

IV. SIMULATIONS RESULTS

For all the following simulations, the burn-in period of the Gibbs sampler has been fixed to $\tau_{beg} = 100$ and the total number of iterations to $\tau_{end} = 1000$. The resampling threshold has been fixed to $N_{threshold} = 0.9$.

A. Application to Bearings-Only Problems With Clutter

To illustrate the MTPF algorithm, we first deal with classical bearings-only problems with three targets. In the context of a slowly maneuvering target, we have chosen a nearly-constant-velocity model (see [22] for a review of the principal dynamical models used in this domain).

1) *Model and the First Scenario Studied:* The state vectors X_t^i represent the coordinates and the velocities in the x - y plane: $X_t^i = (x_t^i, y_t^i, vx_t^i, vy_t^i)$ for $i = 1, 2, 3$. For each target, the discretized state equation associated with time period Δt is

$$X_{t+\Delta t}^i = \begin{pmatrix} I_{2 \times 2} & \Delta t I_{2 \times 2} \\ 0 & I_{2 \times 2} \end{pmatrix} X_t^i + \begin{pmatrix} \frac{\Delta t^2}{2} I_{2 \times 2} \\ \Delta t I_{2 \times 2} \end{pmatrix} V_t \quad (25)$$

where $I_{2 \times 2}$ is the identity matrix in dimension 2, and V_t is a Gaussian zero-mean vector with covariance matrix $\Sigma_V = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix}$. Let \hat{X}_t^i be the estimate of X_t^i computed by the particle filters with $g(x) = x$, i.e., $\hat{X}_t^i = \sum_{n=1}^N q_t^n \hat{x}_t^{n,i}$.

A set of m_t measurements is available at discrete times and can be divided in two subsets.

- A subset of “true” measurements that follow (26). A measurement produced by the i th target is generated according to

$$Y_t = \arctan \left(\frac{y_t^i - y_t^{obs}}{z_t^i - x_t^{obs}} \right) + W_t \quad (26)$$

where W_t is a zero-mean Gaussian noise with covariance σ_w^2 , independent of V_t , and x_{obs} and y_{obs} are the Cartesian coordinates of the observer, which are known. We assume that the measurement produced by one target is available with a detection probability P_d .

- A subset of “false” measurements whose number follows a Poisson distribution with mean λV , where λ is the mean number of false alarms per unit volume. We assume these false alarms are independent and uniformly distributed within the observation volume V .

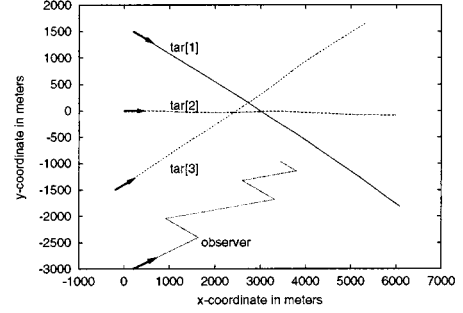


Fig. 4. Trajectories of the three targets and of the observer for experiments in Section IV-A.

The initial positions of the targets and of the observer are the following:

$$X_0^1 = \begin{pmatrix} 200 \text{ m} \\ 1500 \text{ m} \\ 1 \text{ ms}^{-1} \\ -0.5 \text{ ms}^{-1} \end{pmatrix}; \quad X_0^2 = \begin{pmatrix} 0 \text{ m} \\ 0 \text{ m} \\ 1 \text{ ms}^{-1} \\ 0 \text{ ms}^{-1} \end{pmatrix}$$

$$X_0^3 = \begin{pmatrix} -200 \text{ m} \\ -1500 \text{ m} \\ 1 \text{ ms}^{-1} \\ 0.5 \text{ ms}^{-1} \end{pmatrix}; \quad X_0^{obs} = \begin{pmatrix} 200 \text{ m} \\ -3000 \text{ m} \\ 1.2 \text{ ms}^{-1} \\ 0.5 \text{ ms}^{-1} \end{pmatrix}.$$

The dynamic noise is a normal zero-mean Gaussian vector with $\sigma_x = \sigma_y = 0.0005 \text{ ms}^{-2}$. We use the same dynamic noise to predict the particle. The observer is following a leg-by-leg trajectory. Its velocity vector is constant on each leg and modified at the following instants so that

$$\begin{pmatrix} vx_{200}^{obs} \\ vy_{200}^{obs} \end{pmatrix} = \begin{pmatrix} -0.6 \\ 0.3 \end{pmatrix}; \quad \begin{pmatrix} vx_{400}^{obs} \\ vy_{400}^{obs} \end{pmatrix} = \begin{pmatrix} 2.0 \\ 0.3 \end{pmatrix}$$

$$\begin{pmatrix} vx_{600}^{obs} \\ vy_{600}^{obs} \end{pmatrix} = \begin{pmatrix} -0.6 \\ 0.3 \end{pmatrix}; \quad \begin{pmatrix} vx_{800}^{obs} \\ vy_{800}^{obs} \end{pmatrix} = \begin{pmatrix} 2.0 \\ 0.3 \end{pmatrix}$$

$$\begin{pmatrix} vx_{900}^{obs} \\ vy_{900}^{obs} \end{pmatrix} = \begin{pmatrix} -0.6 \\ 0.3 \end{pmatrix}. \quad (27)$$

The trajectories of the three targets and of the observer are represented in Fig. 4, where the arrows indicate the direction of the motion. The measurements are simulated as follows.

- For each target, a “true” bearing is simulated with a probability $P_d = 0.9$ using a Gaussian noise of standard deviation $\sigma_w = 0.05$ rad (about 3°) every time period, i.e., every 6 s.
- The Poisson density mean used is varying between 0 (no clutter) and 3. We consider a total observation volume, i.e., the interval $[-\pi; \pi]$.

The measurement sets simulated for clutter parameters equal to 0, 1, and 3 are plotted in Fig. 5. In this scenario, the data association is particularly difficult: The differences between two bearings issued from two different targets is often lower than the standard deviation of the observation noise.

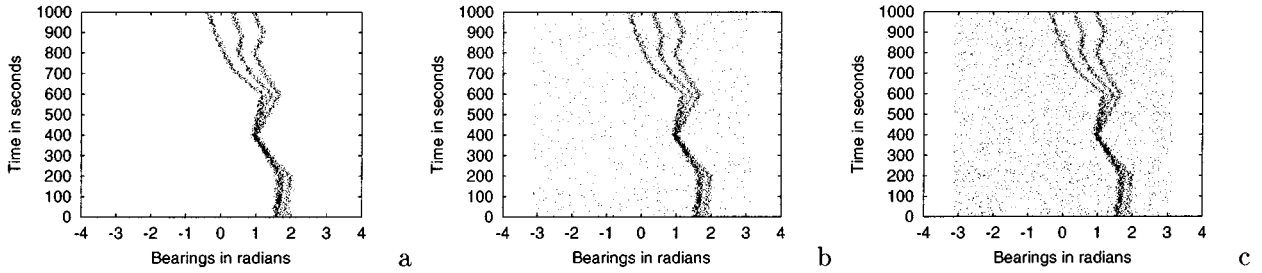


Fig. 5. Measurements simulated with a detection probability $P_d = 0.9$. (a) $\lambda V = 0$. (b) $\lambda V = 1$. (c) $\lambda V = 3$.

2) *Results of the MTPF*: First, the initialization of the particle set has been done according to a Gaussian law whose mean vector X_{mean} and covariance matrix X_{cov} are

$$\begin{aligned} X_{mean}^1 &= X_0^1 + \begin{pmatrix} -10 \text{ m} \\ 100 \text{ m} \\ 0 \text{ ms}^{-1} \\ -0 \text{ ms}^{-1} \end{pmatrix} \\ X_{mean}^2 &= X_0^2 + \begin{pmatrix} -10 \text{ m} \\ -100 \text{ m} \\ 0 \text{ ms}^{-1} \\ 0 \text{ ms}^{-1} \end{pmatrix} \\ X_{mean}^3 &= X_0^3 + \begin{pmatrix} 20 \text{ m} \\ 100 \text{ m} \\ 0 \text{ ms}^{-1} \\ 0 \text{ ms}^{-1} \end{pmatrix} \end{aligned} \quad (28)$$

and

$$X_{cov}^i = \begin{pmatrix} 20 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 0.05 & 0 \\ 0 & 0 & 0 & 0.05 \end{pmatrix}, \quad \text{for } i = 1, 2, 3. \quad (29)$$

To evaluate the performance of the algorithm according to the clutter density, we have performed $P = 20$ different runs of the MTPF with 1000 particles for $\lambda V = 0, 1, 2, 3$. The scenario is the same for each run, i.e., the true target trajectories and the simulated measurements are identical. At each time t , the bias and the standard deviation for the l th component of X^i are defined by

$$\begin{aligned} bias_{t,l} &= \frac{1}{P} \sum_{p=1}^P (\hat{X}_{t,l}^{i,p} - X_{t,l}^i) \\ std_{t,l} &= \sqrt{\frac{1}{P} \sum_{p=1}^P \hat{X}_{t,l}^{i,p2} - \left(\frac{1}{P} \sum_{p=1}^P \hat{X}_{t,l}^{i,p} \right)^2}. \end{aligned} \quad (30)$$

To avoid the compensation of elementary bias of opposite signs, we average the absolute values of the bias $bias_{t,l}$. Then, we define, for each target, and for the l th component

$$bias_l = \frac{1}{T} \sum_{t=1}^T |bias_{t,l}| \quad (31)$$

and we average the standard deviations

$$std_l = \frac{1}{T} \sum_{t=1}^T std_{t,l}. \quad (32)$$

These different quantities, which are normalized by their values obtained with no clutter, are plotted against the clutter parameter in Fig. 6. Except for the x and vx components of the third target, the standard deviation is not very sensitive to clutter. In Fig. 7, the MTPF estimate averaged over the 20 runs have been plotted with the 2σ confidence ellipsoid on position given by $\sigma = std_{t,l}$. In particular, the x component of the third target seems well estimated, which counterbalances the variations observed in Fig. 6.

The ellipsoids plotted in Fig. 7 represent the variance over the 20 runs of the posterior mean estimates and enable us to assess the variance of the MTPF estimator for $N = 1000$ particles. The posterior covariance of the estimate from one particular run is also a useful indicator to assess the quality of the estimation. The 2σ confidence ellipsoids corresponding to the covariance of the posterior estimate are presented for one particular run in Fig. 8(a). As the covariance of dynamic noise is not very high and especially as the prior at time zero is narrow, one might think that the estimates obtained without measurements² could be as good. However, the posterior covariance obtained without using the measurements increase a lot as presented in Fig. 8(b).

With a Pentium III 863 MHz, $N = 1000$ particles, a burn-in period $\tau_{beg} = 100$, and a total amount $\tau_{end} = 1000$ of iterations in the Gibbs sampler, it takes around 1 s per time step to compute the MTPF estimate of three targets with bearings-only measurements.

The next section shows the ability of the MTPF to recover from a poor initialization.

3) *Effect of a Highly Shifted Initialization*: The initial positions and velocities of the objects are the same as in the previous section. The observer is still following a leg-by-leg trajectory, but its initial position is now

$$X_0^{obs} = \begin{pmatrix} 3500 \text{ m} \\ -2500 \text{ m} \\ -0.5 \text{ ms}^{-1} \\ 0 \text{ ms}^{-1} \end{pmatrix}.$$

Its velocity vector is constant on each leg and modified at the following instants so that

$$\begin{aligned} \begin{pmatrix} vx_{30}^{obs} \\ vy_{30}^{obs} \end{pmatrix} &= \begin{pmatrix} 1.2 \\ 0.3 \end{pmatrix}; \quad \begin{pmatrix} vx_{100}^{obs} \\ vy_{100}^{obs} \end{pmatrix} = \begin{pmatrix} -3.0 \\ 0.3 \end{pmatrix} \\ \begin{pmatrix} vx_{200}^{obs} \\ vy_{200}^{obs} \end{pmatrix} &= \begin{pmatrix} 1.2 \\ 0.3 \end{pmatrix}; \quad \begin{pmatrix} vx_{500}^{obs} \\ vy_{500}^{obs} \end{pmatrix} = \begin{pmatrix} -3.0 \\ 0.3 \end{pmatrix} \end{aligned}$$

²Such estimates are obtained by applying the prediction step and by giving constant weights to the particles instead of computing them given the measurements.

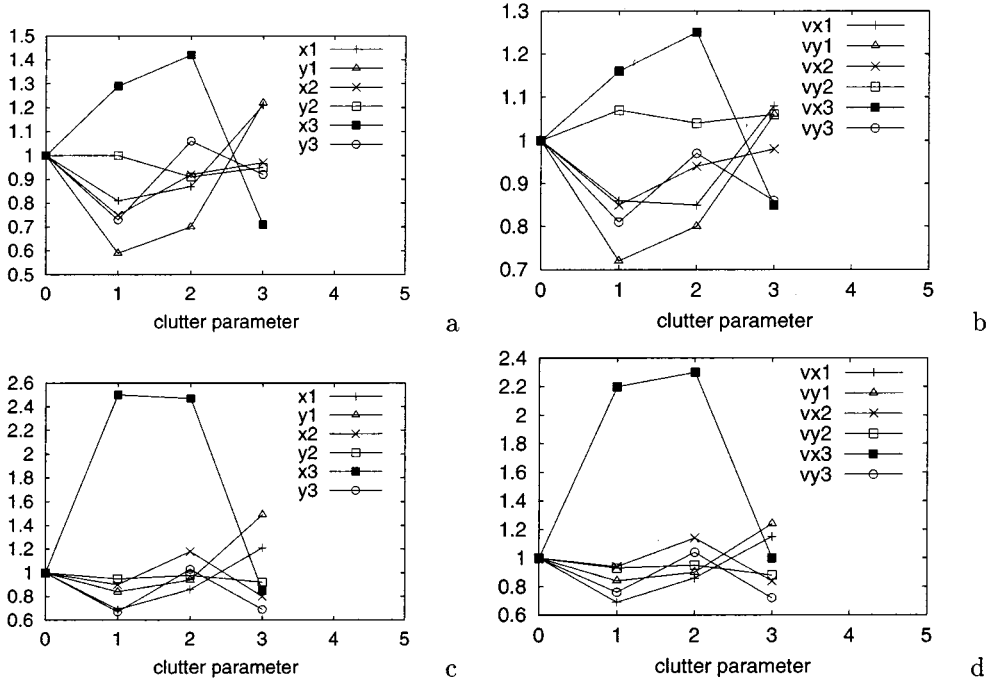


Fig. 6. Bias, resp. std, for clutter parameter = 1, 2, 3 over bias, resp. std, obtained with no clutter obtained with 1000 particles for 20 runs. (a) Bias on x and y position for the three targets. (b) Bias on v_x and v_y position for the three targets. (c) std on x and y position for the three targets. (d) std on v_x and v_y position for the three targets.

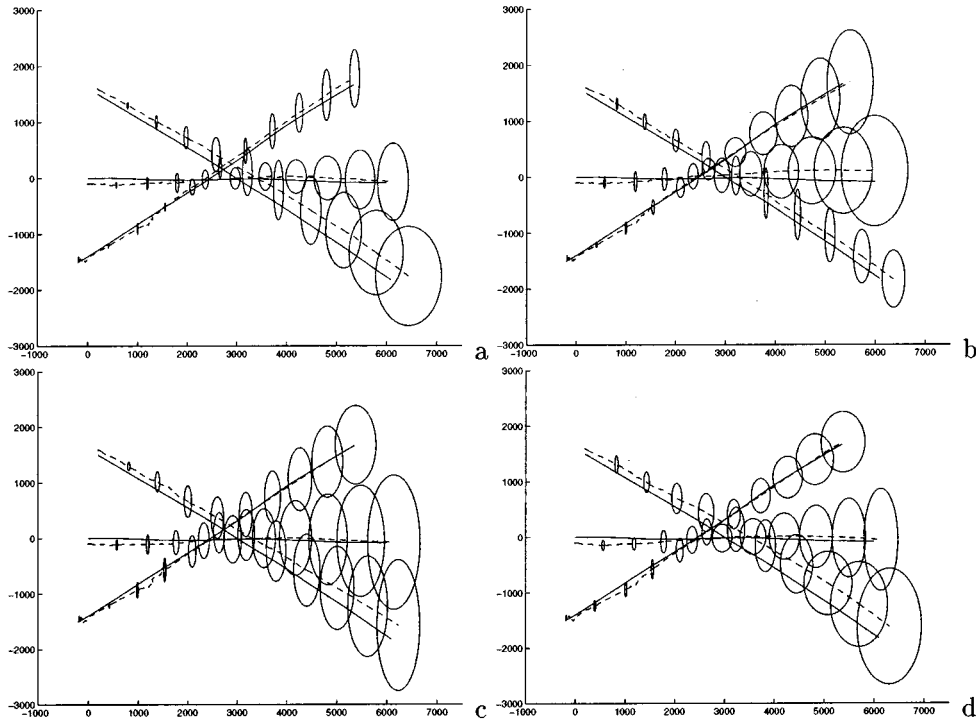


Fig. 7. Averaged MTPF estimate of the posterior means (dashed lines) over 20 runs and associated 2σ confidence ellipsoids for the three targets with $N = 1000$ particles and with a detection probability $P_d = 0.9$. (a) $\lambda V = 0$. (b) $\lambda V = 1$. (c) $\lambda V = 2$. (d) $\lambda V = 3$. The solid lines are the true trajectories.

$$\begin{pmatrix} vx_{600}^{obs} \\ vy_{600}^{obs} \end{pmatrix} = \begin{pmatrix} 1.2 \\ 0.3 \end{pmatrix} \quad \begin{pmatrix} vx_{800}^{obs} \\ vy_{800}^{obs} \end{pmatrix} = \begin{pmatrix} -4.0 \\ 0.3 \end{pmatrix} \\
 \begin{pmatrix} vx_{900}^{obs} \\ vy_{900}^{obs} \end{pmatrix} = \begin{pmatrix} 1.2 \\ 0.3 \end{pmatrix}. \quad (33)$$

The trajectories of the three targets and of the observer are represented in Fig. 9. Compared with the previous section, the first maneuver occurs earlier to make the targets resolvable earlier. The initialization of the particle set has been done according to a Gaussian law whose mean vector X_{mean} and covariance ma-

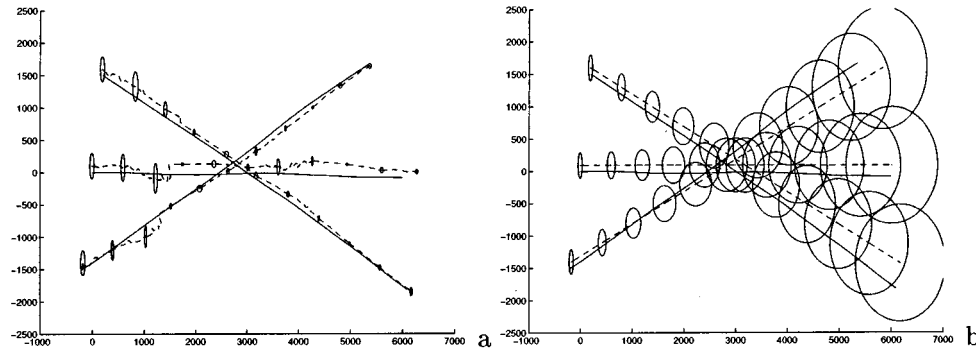


Fig. 8. Single run performance with $N = 1000$ particles. (a) MTPF estimate (dashed lines) and 2σ confidence ellipsoids of the posterior means with a detection probability $P_d = 0.9$ and $\lambda V = 0$. (b) Estimate obtained without measurements and 2σ confidence ellipsoids. The solid lines are the true trajectories.

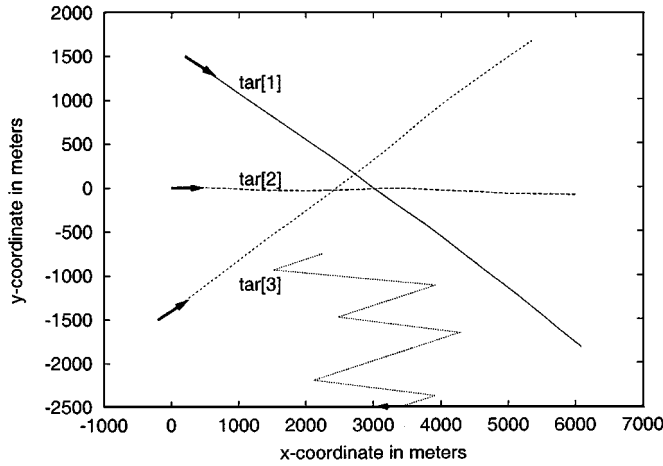


Fig. 9. Trajectories of the three targets and of the observer for experiments in Section IV-A3.

trix X_{cov} are

$$\begin{aligned}
 X_{mean}^1 &= X_0^1 + \begin{pmatrix} 200 \text{ m} \\ 200 \text{ m} \\ 0 \text{ ms}^{-1} \\ 0 \text{ ms}^{-1} \end{pmatrix} \\
 X_{mean}^2 &= X_0^2 + \begin{pmatrix} -200 \text{ m} \\ -200 \text{ m} \\ 0 \text{ ms}^{-1} \\ 0 \text{ ms}^{-1} \end{pmatrix} \\
 X_{mean}^3 &= X_0^3 + \begin{pmatrix} -100 \text{ m} \\ 200 \text{ m} \\ 0 \text{ ms}^{-1} \\ 0 \text{ ms}^{-1} \end{pmatrix}
 \end{aligned} \quad (34)$$

and

$$X_{cov}^i = \begin{pmatrix} 150^2 & 0 & 0 & 0 \\ 0 & 150^2 & 0 & 0 \\ 0 & 0 & 0.1^2 & 0 \\ 0 & 0 & 0 & 0.1^2 \end{pmatrix}, \quad \text{for } i = 1, 2, 3. \quad (35)$$

The measurements have been simulated without any clutter and with a Gaussian noise of standard deviation $\sigma_w = 0.02$ (about 1.15°). The averaged estimates obtained with the MTPF over 20 runs are presented in Fig. 10(a) with the corresponding 2σ ellipsoids. The particles recover very quickly from their shifted

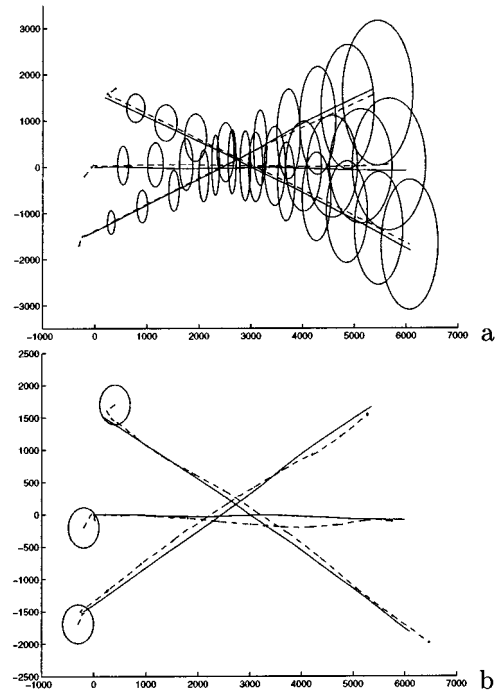


Fig. 10. (a) Averaged MTPF estimate of the posterior means (dashed lines) over 20 runs and associated 2σ confidence ellipsoids for the three targets with $N = 1000$ particles and with a detection probability $P_d = 0.9$ and $\lambda V = 0$. (b) MTPF estimate for one particular run (dashed lines) and 2σ confidence ellipsoids. The solid lines are the true trajectories.

initialization. As in the previous section, the posterior covariance of each MTPF estimate rapidly decreases as shown for one particular run in Fig. 10(b). As the initialization is really far from the true initial positions, the estimates obtained without using any measurements are not acceptable. They are presented in Fig. 11.

4) *Estimation With a JPDAF-EKF Algorithm:* We have compared the MTPF with a classical association algorithm: the JPDAF [3]. Since the measurement equation (26) is nonlinear, we cannot use a classic Kalman filter. According to [1], the modified polar coordinates extended Kalman filter (EKF) gives the best performance among the four single-filter trackers compared. This is the one we used in the JPDAF (see [16], [17], and [27] for details on the implementation of the modified polar coordinates EKF).

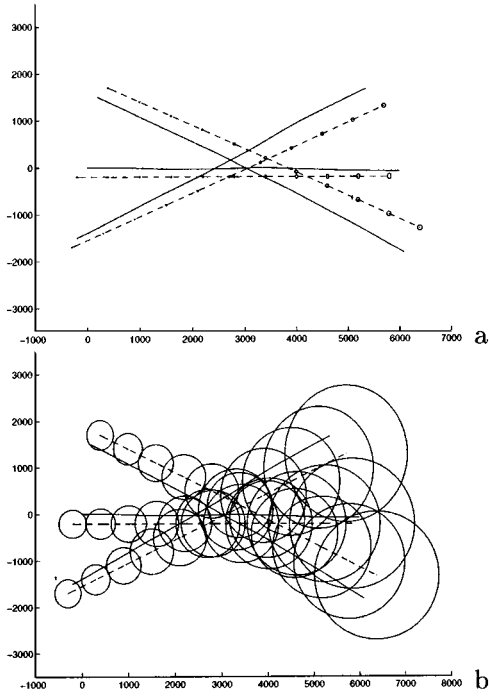


Fig. 11. (a) Averaged estimate obtained without any measurements (dashed lines) over 20 runs and associated 2σ confidence ellipsoids for the three targets with $N = 1000$ particles. (b) Estimate obtained without any measurements for one particular run (dashed lines) and 2σ confidence ellipsoids. The solid lines are the true trajectories.

The validation test of the JPDAF has been performed with a validation probability $P_g = 0.99$. The generation of all the feasible association matrices has been implementing according to a depth-first search algorithm [34]. A JPDAF has then been performed using the simulated bearings of the scenario presented in Section IV-A1 with the clutter density $\lambda V = 1$. Results obtained in this case with the MTPF are presented in Fig. 7(b). As already noted, the scenario is particularly difficult as the measurements are very close. For instance, the second measurement set contains four measurements that have been simulated such that $K_t^j = j$ for $j = 1, \dots, 3$ and $K_t^4 = 0$. The associated likelihood matrix is

$$L = \begin{pmatrix} 1.7836 & 1.6928 & 0.8758 \\ 1.6724 & 1.7808 & 1.2863 \\ 0.4387 & 0.6927 & 1.5691 \\ 0.9852 & 1.3055 & 1.7784 \end{pmatrix} \quad (36)$$

where $L(i, j) = p(Y_t^j = y_t^j | K_t^j = i, X_t^i = x_t^i)$. The validation matrix associated with L using a validation probability $P_g = 0.99$ is $V = I_{4 \times 3}$, which means that the gating does not suppress any measurements. Fig. 12 shows the true trajectories simulated during 1000 time periods and the estimate obtained with the JPDAF. For each object, a given mark indicates the true and estimate positions every 100 times. Note that at time 0, the estimate and the true positions are the same and are indicated by one common mark.

At the end of the scenario, the estimate for target 3 has clearly deteriorated because of the false alarms; some of them are validated as true. A major difference between classical algorithms based on Kalman filtering and particle filtering methods is highlighted here. For a single process X_t to estimate, knowing the state equation (1) and the measurement model (2) with Gaussian

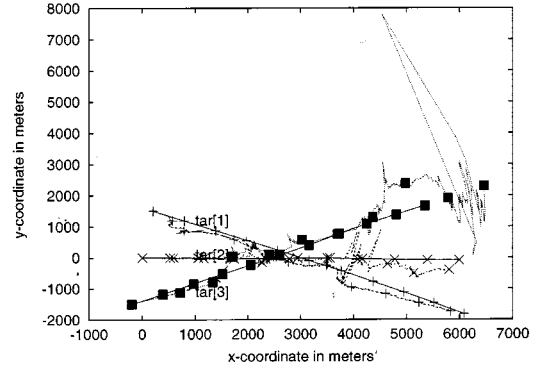


Fig. 12. JPDAF-MPEKF estimates (dotted lines). The solid lines stand for the real trajectories. The true and estimated positions at times 0, 100, and 200 are marked with “+” for target 1, “x” for target 2, and “■” for target 3.

noises, Kalman filtering first computes the predicted estimate according to the following equation:

$$X_{t|t-1}^{kf} = F_t(\hat{X}_{t-1}^{kf}, 0). \quad (37)$$

The measurements are then taken into account through the measurement innovation, i.e., the difference between the measurements and the predicted measurements. The measurement innovation, multiplied by the filter gain K_t , is added to the predicted state

$$\hat{X}_t^{kf} = X_{t|t-1}^{kf} + K_t(Y_t - H_t(X_{t|t-1}^{kf}, 0)). \quad (38)$$

Hence, a high gain or a high measurement noise imply a drastic discrepancy between the predicted and the updated state. Note that the densities of the random variables are no longer Gaussian if F_t or H_t are not linear. In this case, the EKF uses the same equations (37) and (38), where K_t is computed with linearizations of F_t and/or H_t around the predicted state and/or predicted measurement. The obtained estimates are no longer optimal.

In the updating step of particle filtering, the weights of the particles are updated according to the measurements, but the predicted positions are not modified. Consequently, if the prediction is correct, few informative measurements do not deteriorate the MTPF estimate, whereas they cause the divergence of the JPDAF-EKF.

5) *Varying Number of Targets*: To study how the removal of a target can be detected, we have simulated a scenario with a detection hole for one target.³ The true trajectories of the three targets and of the observer are the same as in Fig. 4(a). Each target produces one measurement at each time period according to (26), except during the time interval [600; 700], where the first target does not produce any measurement, and the second produces two measurements y^1 and y^2 according to (26). One particular run of the particle filter with 1000 particles is presented in Fig. 13(a). The differences between the three pairs of bearings simulated are plotted in Fig. 14. The plot of the three estimated trajectories shows that the difficulty of data association has been overcome. Fig. 15 shows the results of the estimation of the three components of π_t , whereas the average of each component π_t^i over successive intervals of 100 time steps and over 20 trials are represented in Fig. 13(b). When there is an

³Note that in the following simulations, where we want to focus only on the issue of varying the number of targets, as well as in those of next section, where we focus on multiple receivers, the clutter is not taken into account.

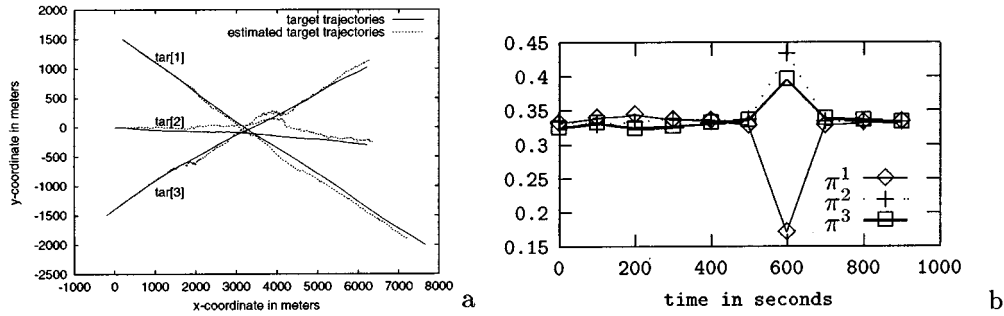


Fig. 13. (a) Target trajectories and their estimate with 1000 particles. (b) Average of the estimated components of the vector π_t over the consecutive ten time intervals of length 100 and over 20 trials.

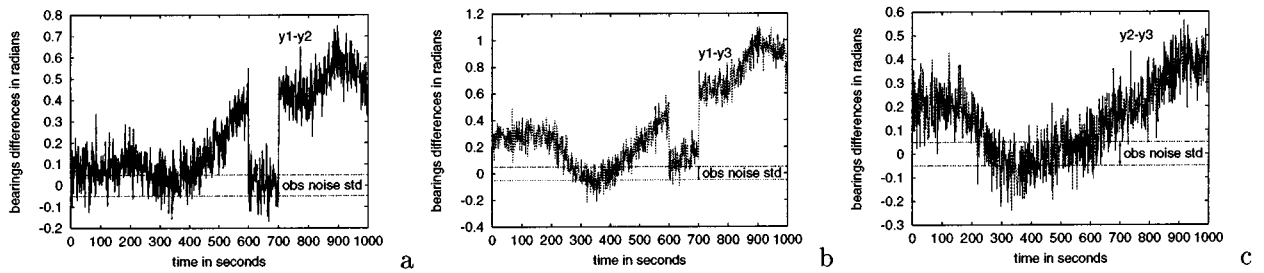


Fig. 14. Differences between the three pairs of target bearings at each time period compared with the standard deviation of the observation noise. (a) Measurements 1 and 2. (b) Measurements 1 and 3. (c) Measurements 2 and 3.

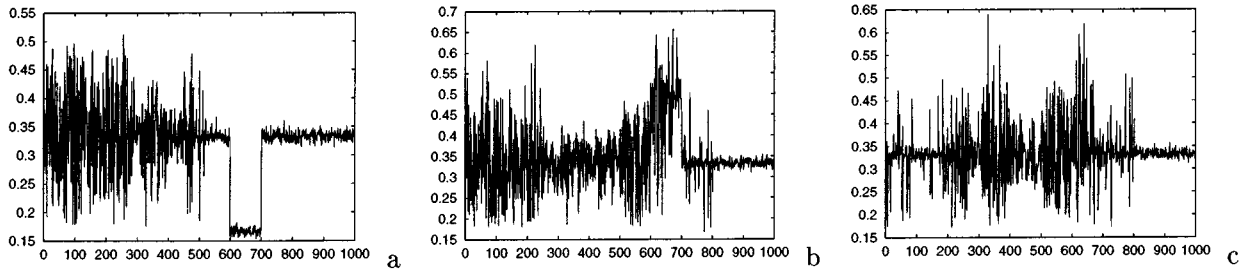


Fig. 15. Estimated components of the vector π_t obtained with 1000 particles. (a) $\hat{\pi}_t^1$. (b) $\hat{\pi}_t^2$. (c) $\hat{\pi}_t^3$.

ambiguity about the origin of the measurements (i.e., when the differences between the bearings are lower than the standard deviation noise), the components of π vary in average around one third for $M = 3$ targets, and they stabilize at uniform estimates (one third for $M = 3$ targets) when the ambiguity disappears. The momentary measurement gap for the first target is correctly handled as the first component π_t^1 is instantaneously estimated as 0.15 from instant 600 to 700.

B. Application to Problems With Active and Passive Measurements

In the following scenario, we consider two targets and one observer whose trajectories are plotted in Fig. 16(a). The initial positions are

$$X_0^1 = \begin{pmatrix} 500 \text{ m} \\ 2000 \text{ m} \\ 0.3 \text{ ms}^{-1} \\ -0.7 \text{ ms}^{-1} \end{pmatrix}; \quad X_0^2 = \begin{pmatrix} 500 \text{ m} \\ -2000 \text{ m} \\ 0.3 \text{ ms}^{-1} \\ 0.7 \text{ ms}^{-1} \end{pmatrix}$$

$$X_0^{obs} = \begin{pmatrix} 600 \text{ m} \\ -7000 \text{ m} \\ 0.5 \text{ ms}^{-1} \\ 0.3 \text{ ms}^{-1} \end{pmatrix}.$$

The difference between the two simulated bearings is very often lower than the measurement noise std as shown in Fig. 16(b). In the following simulations, all the particle clouds have been initialized around the true positions with the covariance matrix X_{cov} defined in (29). In addition, the observer does not follow a leg-by-leg trajectory. This makes the estimation of the trajectories quite difficult, and a lot of runs of the MTPF lost the track. Consequently, the standard deviation over 20 runs increases a lot through time, as illustrated by Fig. 16(d). To improve tracking performance, we study the impact of adding active measurements (here, ranges). We assume that noisy ranges are available periodically during time intervals of length T with period P , i.e., these measurements are present at time t if $t \bmod P \in [0; T]$. A noisy range associated with the i th target is supposed to follow the equation

$$R_t^i = \sqrt{(x_t^i - x_t^{obs})^2 + (y_t^i - y_t^{obs})^2} + Z_t \quad (39)$$

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 $\sigma_z = 10^{-6}$. This noise modeling seems more realistic than the constant standard deviation modeling generally used in such contexts. For instance, for $T = 30$ and $P = 100$, the simulated ranges of the two targets are shown

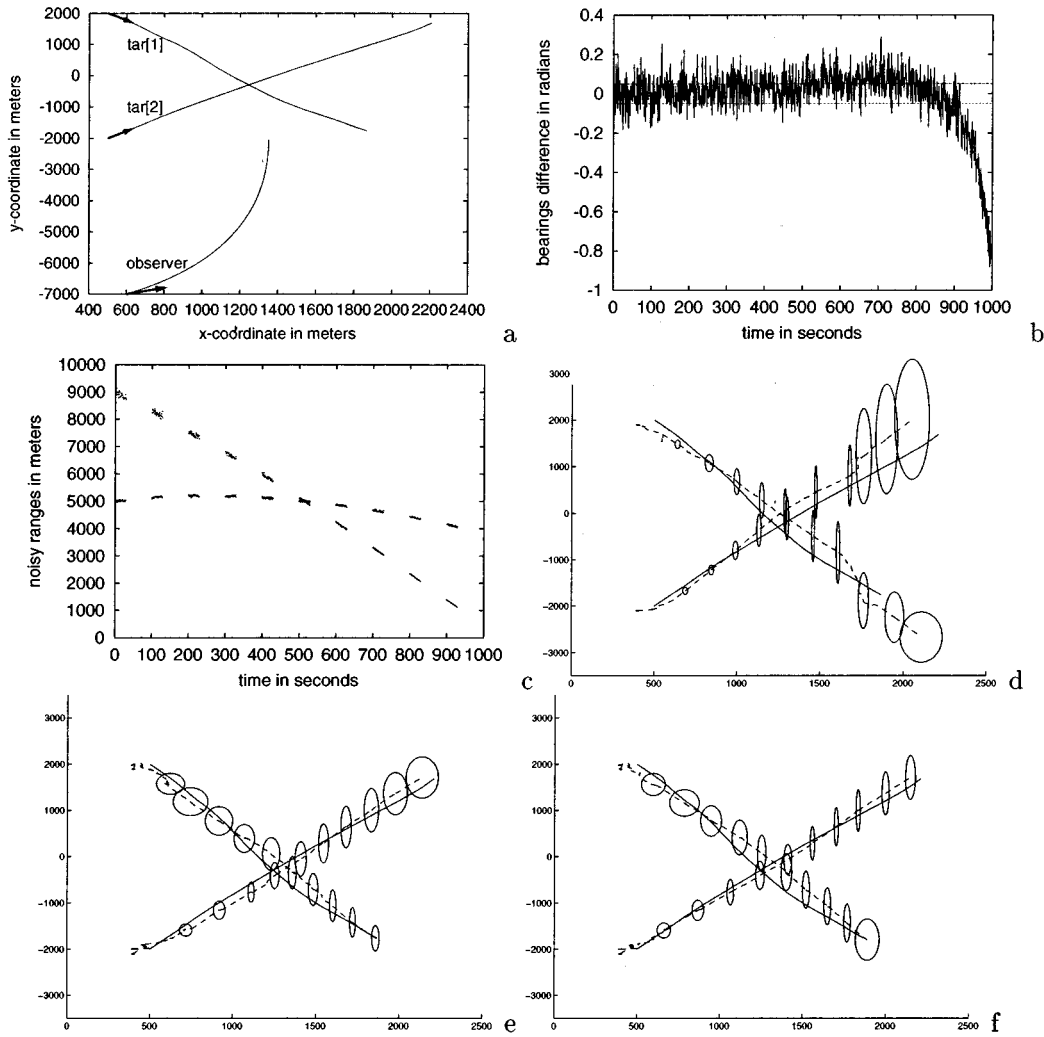


Fig. 16. (a) Trajectories of the targets and of the observer. (b) Difference between the noisy bearings associated with the targets compared with the standard deviation of the measurement noise = 0.05, i.e., 2.8°. (c) Noisy ranges simulated for $T = 30$ and $P = 100$. (d)–(f) Averaged estimates (dashed lines) and 2σ confidence ellipsoids obtained with bearings measurements and 0%, 20%, and 50% of range measurements, respectively. The solid lines stand for the real trajectories.

in Fig. 16(c). The evolution of the bias and the standard deviation of the estimation errors has been studied according to the quantity of active measurements on the one hand and to their temporal distribution on the other.

1) *Quantity of Active Measurements:* For these experimentations, we have fixed $P = 100$ and taken $T = 0, 10, \dots, 100$. Fig. 17 summarizes the evolution of the bias and the standard deviation of the estimation errors as a function of the active measurement percentage. Fig. 16(e) and (f) shows the MRMPF estimated posterior means averaged over 20 runs and the 2σ confidence ellipsoids with, respectively, 20% and 50% of active measurements.

First, the addition of active measurements particularly improves the estimation of the components y and vy for the two targets. Fig. 16(d)–(f) also shows the drastic reduction of the size of the confidence ellipsoids along the y -axis when range measurements are added. The x and vx -positions of the two targets are actually very close, and the bearings measurements do not help to dissociate the targets because of the difficulty of data association. However, as the difference between the y -positions

of the two targets is very high, the range measurements are very different. They thus help a lot to distinguish the targets and to solve the data association.

The percentage of 20% of active measurements appears to be a good compromise between a significant improvement of the estimation and a reasonable quantity of active measurements.

With a Pentium III 863 MHz, $N = 1000$ particles, a burn-in period $\tau_{beg} = 100$, and a total amount $\tau_{end} = 1000$ of iterations in the Gibbs sampler, it takes around 840 ms per time step to compute the MTPF estimates of two targets with bearings measurements and 20% of range measurements.

2) *Temporal Distribution of Active Measurements:* We now look at the impact of the temporal distribution of the active measurements: the ratio of passive over active measurements is fixed to 5 (i.e., to 20% of active measurements). The interval lengths considered are $T = 10, 20, 40$, and 100. The averaged MRMPF estimates and the 2σ confidence ellipsoids obtained with 20 runs and 1000 particles are represented in Fig. 18 for different T values. First of all, if the state evolution was deterministic, the better choice would be to consider active measure-

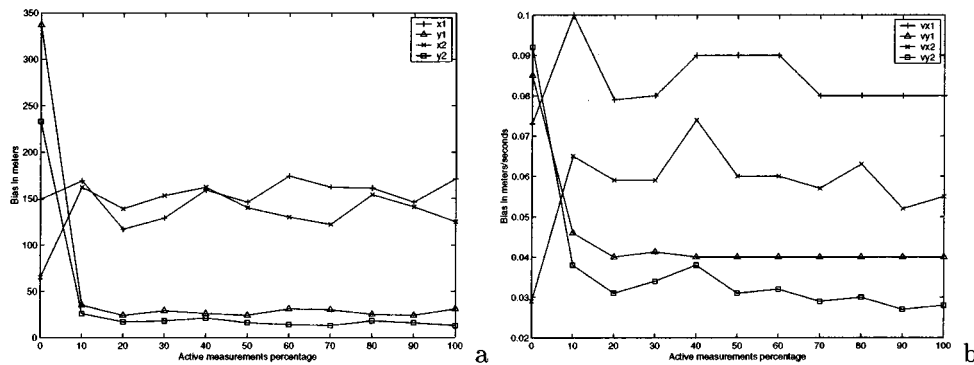


Fig. 17. Bias on the estimation of the hidden states (x, y, v_x, v_y) of the two targets with 1000 particles over 20 runs. (a) Bias on x and y . (b) Bias on v_x and v_y .

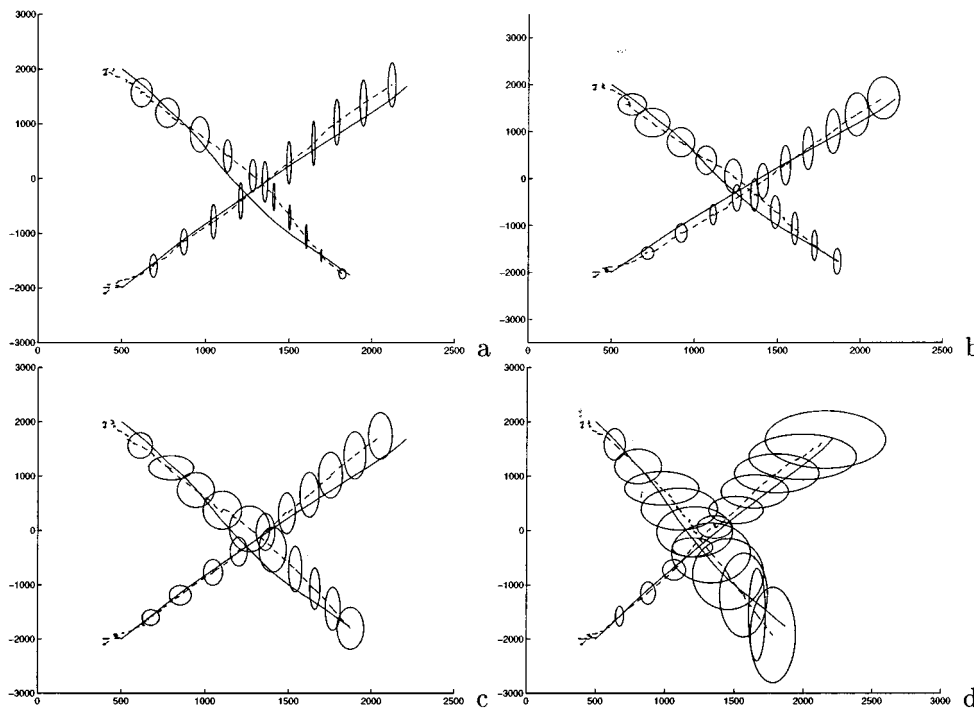


Fig. 18. Averaged MRMPF estimates (dotted lines) and 2σ confidence ellipsoids (dashed lines) with 1000 particles: (a) $T = 10, P = 50$; (b) $T = 20, P = 100$; (c) $T = 40, P = 200$; (d) $T = 100, P = 500$. The solid lines are the true trajectories.

ments at the beginning and at the end of the scenario. In our case, the state evolution is stochastic. We observe that the bias is independent of the temporal distribution of range measurements. The size of the 2σ confidence ellipsoids increases with T . The active measurements should then be available as frequently as possible to improve the estimation performance.

V. CONCLUSION

Two major extensions of the classical particle filter have been presented in order to deal first with multiple targets (MTPF) and then with multiple receivers (MRMTPF). Considering the data association from a stochastic point of view, Gibbs sampling is the workhorse for estimating association vectors, thus avoiding combinatorial drawbacks. Moreover, the particle filtering performs satisfactorily, even in the presence of dense clutter. A next step would be to deal with more realistic clutter models. Two statistical tests have also been proposed for detecting changes

of the target states (emitting or not). Even if the MTPF is quite versatile, it can suffer from initialization problems. This drawback cannot be completely avoided in the multitarget context. This will be addressed in future studies. Finally, MTPF has been extended to multiple receivers and multiple measurements (here passive and active). In this context, the effects of the temporal distribution of active measurement have been investigated. Preliminary results on this aspect show all the importance of measurement scheduling.

REFERENCES

- [1] S. Arulampalam, "A comparison of recursive style angle-only target motion analysis algorithms," Electron. Surveillance Res. Lab., Salisbury, Australia, Tech. Rep. DSTO-TR-0917, Jan. 2000.
- [2] D. Avitzour, "Stochastic simulation Bayesian approach to multitarget approach," *Proc. Inst. Elect. Eng.—Radar, Sonar, Navigat.*, vol. 142, no. 2, pp. 41–44, Apr. 1995.
- [3] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*. New York: Academic, 1988.

- [4] G. Casella and E. I. George, "Explaining the Gibbs sampler," *Amer. Statist.*, vol. 46, no. 3, pp. 167–174, 1992.
- [5] G. Celeux and J. Diebolt, "A stochastic approximation type EM algorithm for the mixture problem," *Stochast. Stochast. Rep.*, vol. 41, pp. 119–134, 1992.
- [6] J. Mac Cormick and M. Isard, "Bramble: A Bayesian multiple-blob tracker," in *Proc. 8th Int. Conf. Comput. Vision*, vol. 2, Vancouver, BC, Canada, July 9–12, 2001, pp. 34–41.
- [7] J. F. G. de Freitas, "Bayesian methods for neural networks," Ph.D. dissertation, Trinity College, Univ. Cambridge, Cambridge, U.K., 1999.
- [8] J. Diebolt and C. P. Robert, "Estimation of finite mixture distributions through Bayesian sampling," *J. R. Statist. Soc. B*, vol. 56, pp. 363–375, 1994.
- [9] A. Doucet, "On sequential simulation-based methods for Bayesian filtering," Signal Process. Group, Dept. Eng., Cambridge, U.K., Tech. Rep. CUED/F-INFENG/TR 310, 1998.
- [10] —, "Convergence of sequential Monte Carlo methods," Signal Process. Group, Dept. Eng., Cambridge, U.K., Tech. Rep. CUED/F-INFENG/TR 381, 2000.
- [11] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE J. Oceanic Eng.*, vol. OE-8, pp. 173–184, July 1983.
- [12] H. Gauvrit, J.-P. Le Cadre, and C. Jauffret, "A formulation of multitarget tracking as an incomplete data problem," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 33, pp. 1242–1257, Oct. 1997.
- [13] A. E. Gelfand and A. F. M. Smith, "Sampling-based approaches to calculating marginal densities," *J. Amer. Statist. Assoc.*, vol. 85, pp. 398–409, 1990.
- [14] N. Gordon, "A hybrid bootstrap filter for target tracking in clutter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 33, pp. 353–358, Jan. 1997.
- [15] N. Gordon, D. Salmond, and A. Smith, "Novel approach to non-linear/non-Gaussian Bayesian state estimation," *Proc. Inst. Elect. Eng. F, Radar Signal Process.*, vol. 140, no. 2, pp. 107–113, April 1993.
- [16] W. Grossman, "Bearings-only tracking: A hybrid coordinate system approach," *J. Guidance, Contr., Dynamics*, vol. 17, no. 3, pp. 451–457, May/June 1994.
- [17] J. C. Hassab, *Underwater Signal and Data Processing*. Boca Raton, FL: CRC, 1989, pp. 221–243.
- [18] C. Hue, J.-P. Le Cadre, and P. Pérez, "Tracking multiple objects with particle filtering," IRISA, Rennes, France, Tech. Rep., Oct. 2000.
- [19] M. Hürzeler and H. R. Künsch, "Monte Carlo approximations for general state space models," *J. Comput. Graph. Statist.*, vol. 7, pp. 175–193, 1998.
- [20] M. Isard and A. Blake, "CONDENSATION—Conditional density propagation for visual tracking," *Int. J. Comput. Vis.*, vol. 29, no. 1, pp. 5–28, 1998.
- [21] A. Kong, J. S. Liu, and W. H. Wong, "Sequential imputation method and Bayesian missing data problems," *J. Amer. Statist. Assoc.*, vol. 89, pp. 278–288, 1994.
- [22] X. Li and V. Jilkov, "A survey of maneuvering target tracking: Dynamic models," in *Proc. Conf. Signal Data Process. Small Targets*, Orlando, FL, Apr. 2000.
- [23] J. S. Liu, "Metropolized independent sampling with comparison to rejection sampling and importance sampling," *Statist. Comput.*, vol. 6, pp. 113–119, 1996.
- [24] J. MacCormick and A. Blake, "A probabilistic exclusion principle for tracking multiple objects," in *Proc. 7th Int. Conf. Comput. Vis.*, Kerkyra, Greece, Sept. 20–27, 1999, pp. 572–578.
- [25] P. Del Moral and A. Guionnet, "Central limit theorem for nonlinear filtering and interacting particle systems," *Ann. Appl. Probab.*, vol. 9, no. 2, pp. 275–297, 1999.
- [26] C. Musso and N. Oudjane, "Particle methods for multimodal filtering," in *Proc. 2nd IEEE Int. Conf. Inform. Fusion*, Silicon Valley, CA, July 6–8, 1999.
- [27] N. Peach, "Bearings-only tracking using a set of range parameterised extended Kalman filters," *Proc. Inst. Elect. Eng. Contr. Theory Appl.*, vol. 142, no. 1, Jan. 1995.
- [28] D. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Automat. Contr.*, vol. 24, no. AC-6, pp. 84–90, 1979.
- [29] D. Schulz, W. Burgard, D. Fox, and A. B. Cremers, "Tracking multiple moving targets with a mobile robot using particle filters and statistical data association," in *Proc. IEEE Int. Conf. Robotics Automat.*, Seoul, Korea, May 21–26, 2001, pp. 1665–1670.
- [30] A. Smith and G. Roberts, "Bayesian computation via the Gibbs Sampler and related Markov chain Monte Carlo methods," *J. R. Statist. Soc. B*, vol. 55, no. 1, pp. 3–24, 1993.
- [31] M. Stephens, "Bayesian methods for mixtures of normal distributions," Ph.D. dissertation, Magdalen College, Oxford Univ., Oxford, U.K., 1997.
- [32] R. L. Streit and T. E. Luginbuhl, "Maximum likelihood method for probabilistic multi-hypothesis tracking," *Proc. SPIE*, vol. 2235, Apr. 5–7, 1994.
- [33] P. Willett, Y. Ruan, and R. Streit, "The PMHT for maneuvering targets," *Proc. SPIE*, vol. 3373, Apr. 1998.
- [34] B. Zhou, "Multitarget tracking in clutter: Fast algorithms for data association," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 29, pp. 352–363, Apr. 1993.

Carine Hue was born in 1977. She received the M.Sc. degree in mathematics and computer science in 1999 from the University of Rennes, Rennes, France. Since 1999, she has been pursuing the Ph.D. degree with IRISA, Rennes, and works on particle filtering methods for tracking in signal processing and image analysis.

Jean-Pierre Le Cadre (M'93) received the M.S. degree in mathematics in 1977 and the "Doctorat de 3eme cycle" degree in 1982 and the "Doctorat d'Etat" degree in 1987, both from INPG, Grenoble, France.

From 1980 to 1989, he was with the Groupe d'Etudes et de Recherche en Detection Sous-Marine (GERDSM), which is a laboratory of the Direction des Constructions Navales (DCN), mainly on array processing. In this area, he conducted both theoretical and practical researches (towed arrays, high-resolution methods, performance analysis, etc.). Since 1989, he has been with IRISA/CNRS, where is "Directeur de Recherche." His interests have now moved toward other topics like system analysis, detection, multitarget tracking, data association, and operations research.

Dr. Le Cadre received (with O. Zugmeyer) the Eurasip Signal Processing best paper award in 1993.

Patrick Pérez was born in 1968. He graduated from Ecole Centrale Paris, France, in 1990. He received the Ph.D. degree in signal processing and telecommunications from the University of Rennes, Rennes, France, in 1993.

After one year as an INRIA post-doctoral fellow at Brown University, Providence, RI, he was appointed a Full Time INRIA researcher. In 2000, he joined Microsoft Research, Cambridge, U.K. His research interests include probabilistic models for image understanding, high-dimensional inverse problems in image analysis, analysis of motion, and tracking in image sequences.