Monte Carlo Filtering for Multi-Target Tracking and Data Association

JACO VERMAAK

SIMON J. GODSILL, Member, IEEE
Cambridge University

PATRICK PÉREZ
IRISA/INRIA Rennes
France

We present Monte Carlo methods for multi-target tracking and data association. The methods are applicable to general nonlinear and non-Gaussian models for the target dynamics and measurement likelihood. We provide efficient solutions to two very pertinent problems: the data association problem that arises due to unlabelled measurements in the presence of clutter, and the curse of dimensionality that arises due to the increased size of the state-space associated with multiple targets. We develop a number of algorithms to achieve this. The first, which we refer to as the Monte Carlo joint probabilistic data association filter (MC-JPDAF), is a generalisation of the strategy proposed in [1] and [2]. As is the case for the JPDAF, the distributions of interest are the marginal filtering distributions for each of the targets, but these are approximated with particles rather than Gaussians. We also develop two extensions to the standard particle filtering methodology for tracking multiple targets. The first, which we refer to as the sequential sampling particle filter (SSPF), samples the individual targets sequentially by utilising a factorisation of the importance weights. The second, which we refer to as the independent partition particle filter (IPPF), assumes the associations to be independent over the individual targets, leading to an efficient component-wise sampling strategy to construct new particles. We evaluate and compare the proposed methods on a challenging synthetic tracking problem.

Manuscript received April 14, 2004; revised September 22, 2004; released for publication November 12, 2004.

IEEE Log No. T-AES/41/1/844826.

Refereeing of this contribution was handled by P. K. Willett.

Authors' addresses: J. Vermaak and S. J. Godsill, Signal Processing Laboratory, Cambridge University, Engineering Dept., Trumpington St., Cambridge, Cambridgeshire CB2 1PZ, UK, E-mail: (jv211@eng.cam.ac.uk); P. Pérez, IRISA/INRIA Rennes, Campus de Beaulieu, 35042 Rennes Cedex, France.

I. INTRODUCTION

The detection and tracking of multiple targets is a problem that arises in a wide variety of contexts. Examples include radar based tracking of aircraft, sonar based tracking of sea animals or submarines, video based identification and tracking of people for surveillance or security purposes, and many more. The most commonly used framework for tracking is that of Bayesian sequential estimation. This framework is probabilistic in nature, and thus facilitates the modelling of uncertainties due to inaccurate models, sensor errors, environmental noise, etc. The general recursions update the posterior distribution of the target state, also known as the filtering distribution, through two stages: a prediction step that propagates the posterior distribution at the previous time step through the target dynamics to form the one step ahead prediction distribution, and a filtering step that incorporates the new data through Bayes' rule to form the new filtering distribution. In theory the framework requires only the definition of a model for the target dynamics, a likelihood model for the sensor measurements, and an initial distribution for the target state.

The application of the Bayesian sequential estimation framework to real world multi-target tracking problems is plagued by two difficulties. First, realistic models for the target dynamics and measurement processes are often nonlinear and non-Gaussian, so that no closed-form analytic expression can be obtained for the tracking recursions. In fact, closed-form expressions are available only in a small number of cases. The most well-known of these arises when both the dynamic and likelihood models are linear and Gaussian, leading to the celebrated Kalman filter (KF) [3]. The second difficulty is due to the fact that in most practical tracking applications the sensors yield unlabelled measurements of the targets. This leads to a combinatorial data association problem that is very challenging when targets have a small separation compared with the measurement errors. Furthermore, clutter measurements may arise due to multi-path effects, sensor errors, spurious objects, etc., further increasing the complexity of the data association problem.

Many strategies have been proposed in the literature to address the difficulties associated with multi-target tracking. We do not attempt to give an exhaustive summary here, but rather highlight some of the key contributions over the years. When tracking a single object closed-form expressions are generally not available for nonlinear or non-Gaussian models, and approximate methods are required. The extended KF (EKF) [3] linearises models with weak nonlinearities around the current state estimate, so that the KF recursions can still be applied. However, the performance of the EKF degrades rapidly as the

nonlinearities become more severe. To alleviate this problem the unscented KF (UKF) [4], [5] maintains the second-order statistics of the target distribution by recursively propagating a set of carefully selected sigma points. This method requires no linearisation, and generally yields more robust estimates. One of the first attempts to deal with models with non-Gaussian state or observation noise is the the Gaussian sum filter (GSF) [6] that works by approximating the non-Gaussian target distribution with a mixture of Gaussians. It suffers, however, from the same shortcoming as the EKF in that linear approximations are required. It also leads to a combinatorial growth in the number of mixture components over time, calling for ad-hoc strategies to prune the number of components to a manageable level. An alternative method for non-Gaussian models that does not require any linear approximations has been proposed in [7]. It approximates the non-Gaussian state numerically with a fixed grid, and applies numerical integration for the prediction step and Bayes' rule for the filtering step. However, the computational cost of the numerical integration grows exponentially with the dimension of the state-space, and the method becomes impractical for dimensions larger than four.

For general nonlinear and non-Gaussian models, particle filtering [8, 9], also known as sequential Monte Carlo (SMC) [10–12], or CONDENSATION [13], has become a practical and popular numerical technique to approximate the Bayesian tracking recursions. This is due to its efficiency, simplicity, flexibility, ease of implementation, and modelling success over a wide range of challenging applications. It represents the target distribution with a set of samples, known as particles, and associated importance weights, which are then propagated through time to give approximations of the target distribution at subsequent time steps. It requires only the definition of a suitable proposal distribution from which new particles can be generated, and the ability to evaluate the dynamic and likelihood models. As opposed to the strategy in [7], the computational complexity for particle filters does not necessarily become prohibitive with an increase in the dimension of the state-space.

A large number of strategies are available to solve the data association problem. These can be broadly categorised as either single frame assignment methods, or multi-frame assignment methods. We focus mainly on the former here. Methods to solve the multi-frame assignment problem include, e.g. Lagrangian relaxation [14]. The multiple hypotheses tracker (MHT) [15] attempts to keep track of all the possible association hypotheses over time. This is an NP-hard problem, since the number of association hypotheses grows exponentially over time. Thus methods are required to reduce the computational complexity. The nearest neighbour standard filter (NNSF) [16] associates each target with the closest measurement in the target space. However, this simple procedure prunes away many feasible hypotheses. In this respect the joint probabilistic data association filter (JPDAF) [16, 17] is more appealing. At each time step infeasible hypotheses are pruned away using a gating procedure. A filtering estimate is then computed for each of the remaining hypotheses, and combined in proportion to the corresponding posterior hypothesis probabilities. The main shortcoming of the JPDAF is that, to maintain tractability, the final estimate is collapsed to a single Gaussian, thus discarding pertinent information. Subsequent work addressed this shortcoming by proposing strategies to instead reduce the number of mixture components in the original mixture to a tractable level [18, 19]. Still, many feasible hypotheses may be discarded by the pruning mechanisms. The probabilistic multiple hypotheses tracker (PMHT) [20, 21] (suboptimally) assumes the association variables to be independent to work around the problems with pruning. It leads to an incomplete data problem that can be efficiently solved using the expectation maximisation (EM) algorithm [22]. However, the PMHT is a batch strategy, and thus not suitable for online applications. The standard version of the PMHT is also generally outperformed by the JPDAF. Some of the reasons for this, and a number of possible solutions, are discussed in [23].

Even though methods to solve the data association problem do not usually rely on linear and Gaussian models, this assumption is often made to simplify hypothesis evaluation for target originated measurements. For example, nonlinear models can be accommodated by suitable linearising using the EKF. As for the EKF, however, the performance of the algorithms degrades as the nonlinearities become more severe. Recently strategies have been proposed to combine the JPDAF with particle techniques to accommodate general nonlinear and non-Gaussian models [1, 2, 24, 25]. The data association problem has also been addressed directly in the context of particle filtering. The feasibility of multi-target tracking with particle filters has first been claimed in [26] and [27], but the examples there deal only with a single target. In [28] a method is described that computes the distribution of the association hypotheses using a Gibbs sampler [29] at each time step. The method is similar in spirit to the one described in [30] that uses Markov chain Monte Carlo (MCMC) techniques [31] to compute the correspondences between image points within the context of stereo reconstruction. The main problem with these MCMC strategies is that they are iterative in nature and take an unknown number of iterations to converge. They are thus not entirely suitable for online applications. In [32] a method is presented where the associations are sampled from an optimally designed importance distribution. The method is

intuitively appealing since the association hypotheses are treated in a similar fashion to the target state, so that the resulting algorithm is noniterative. It is, however, restricted to jump Markov linear systems (JMLS) [33]. An extension of this strategy, based on the auxiliary particle filter (APF) [34] and the UKF, that is applicable to general jump Markov systems (JMS) is presented in [35]. Another approach that is similar in spirit is described in [36]. Samples for the association hypotheses are generated from an efficient proposal distribution based on the notion of a soft-gating of the measurements.

Particle filters have also been applied to the problem of multi-target tracking based on raw measurements, e.g. [37]–[39]. These so-called track before detect (TBD) strategies construct a generative model for the raw measurements in terms of a multi-target state hypothesis, thus avoiding an explicit data association step. However, such measurements are not always readily available in practical systems, and may lead to a larger computational complexity if they are. For these reasons we do not consider TBD techniques here, but instead focus our attention on the more conventional thresholded measurement procedure.

The multi-target tracking problem suffers from the curse of dimensionality. As the number of targets increases, the size of the joint state-space increases exponentially. If care is not taken in the design of proposal distributions an exponentially increasing number of particles may be required to cover the support of the multi-target distribution and maintain a given level of accuracy. For poor proposals it may commonly occur that particles contain a mixture of good estimates for some target states, and bad estimates for other target states. In the computation of the particle weights, however, the entire particle is penalised for the component targets with bad estimates, so that even components with good estimates are destroyed during the resampling stage. This leads to a rapid depletion of the Monte Carlo representation. This problem has been acknowledged before in [40], where a selection strategy is proposed that constructs new particles from individual target states that are deemed to be accurately estimated. We consider similar proposal strategies here.

Here we present a number of particle-filter-based strategies for multi-target tracking and data association for general nonlinear and non-Gaussian models. The first, which we refer to as the Monte Carlo joint probabilistic data association filter (MC-JPDAF), is a generalisation of the strategy proposed in [1] and [2] to multiple observers and arbitrary proposal distributions. As is the case for the JPDAF, the distributions of interest are the marginal filtering distributions for each of the targets. Contrary to the JPDAF, which approximates these with Gaussians, we use particle approximations. The marginal association probabilities required during the filtering step are computed using these particles. We also develop two extensions to the standard particle filtering methodology for tracking multiple targets. The first is an exact strategy that samples the individual targets sequentially by utilising a factorisation of the importance weights. We refer to this algorithm as the sequential sampling particle filter (SSPF). The second strategy assumes the associations to be independent over the individual targets. This is similar to the approximation made in the PMHT, and implies that measurements can be assigned to more than one target. This assumption effectively removes all dependencies between the individual targets, leading to an efficient component-wise sampling strategy to construct new particles. We refer to this approach as the independent partition particle filter (IPPF). As opposed to the JPDAF, neither approach requires a gating procedure, but captures the notion of a soft-gating of the measurements by an efficiently designed proposal distribution, similar in spirit to the one in [36].

The remainder of the paper is organised as follows. Section II introduces the Bayesian sequential estimation framework, and shows how the general recursions can be implemented using Monte Carlo techniques, leading to the particle filter. Section III discusses the modelling assumptions for the multi-target tracking problem. It develops models for the target dynamics, discusses the data association problem, and derives a likelihood model conditional on a known association hypothesis. It also develops a prior for the association hypothesis, which is required by the subsequent tracking algorithms. The MC-JPDAF is outlined in Section IV. The general framework is first presented, together with the required assumptions, and then it is shown how the framework can be implemented using Monte Carlo techniques. The multi-target particle filtering algorithms are developed in Section V, together with efficient proposals for the association hypothesis. Section VI discusses some choices for the target state proposal distribution that is applicable to both the MC-JPDAF and the multi-target particle filtering algorithms. The proposed methods are evaluated and compared on a challenging synthetic tracking problem in Section VII. Finally, Section VIII summarises the paper and makes a number of suggestions for future research.

II. BAYESIAN SEQUENTIAL ESTIMATION AND PARTICLE FILTERS

Since particle filters will form the core component of the multi-target tracking algorithms developed here, we begin by a brief description of the Bayesian sequential estimation framework and its Monte Carlo approximation, i.e., the particle filter. We describe the framework for a generic model parameterised by a

state $\mathbf{x}_t$, where t denotes the discrete time index. For tracking the distribution of interest is the posterior $p(\mathbf{x}_t \mid \mathbf{y}_{1:t})$, also known as the filtering distribution, where $\mathbf{y}_{1:t} = (\mathbf{y}_1 \dots \mathbf{y}_t)$ denotes all the observations up to the current time step. In the Bayesian sequential estimation framework the filtering distribution can be computed according to the two step recursion

prediction step:

$$p(\mathbf{x}_t \mid \mathbf{y}_{1:t-1}) = \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) p(\mathbf{x}_{t-1} \mid \mathbf{y}_{1:t-1}) d\mathbf{x}_{t-1}$$

filtering step:

$$p(\mathbf{x}_t \mid \mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_t \mid \mathbf{x}_t) p(\mathbf{x}_t \mid \mathbf{y}_{1:t-1})}{p(\mathbf{y}_t \mid \mathbf{y}_{1:t-1})} \tag{1}$$

where the prediction step follows from marginalisation, and the new filtering distribution is obtained through a direct application of Bayes' rule. The recursion requires the specification of a dynamic model describing the state evolution $p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$, and a model for the state likelihood in the light of the current measurements $p(\mathbf{y}_t \mid \mathbf{x}_t)$.[1] The recursion is initialised with some distribution for the initial state $p(\mathbf{x}_0)$. Once the sequence of filtering distributions is known point estimates of the state can be obtained according to any appropriate loss function, leading to, e.g., maximum a posteriori (MAP) and minimum mean square error (MMSE) estimates.

The tracking recursion yields closed-form expressions in only a small number of cases. The most well-known of these is the KF [3] for linear and Gaussian dynamic and likelihood models. For general nonlinear and non-Gaussian models the tracking recursion becomes analytically intractable, and approximation techniques are required. SMC methods [10–12], otherwise known as particle filters [8, 9], or CONDENSATION [13], have gained a lot of popularity in recent years as a numerical approximation strategy to compute the tracking recursion for complex models. This is due to their efficiency, simplicity, flexibility, ease of implementation, and modelling success over a wide range of challenging applications.

The basic idea behind particle filters is very simple. Starting with a weighted set of samples $\{w_{t-1}^{(n)}, \mathbf{x}_{t-1}^{(n)}\}_{n=1}^{N}$ approximately distributed according to $p(\mathbf{x}_{t-1} \mid \mathbf{y}_{1:t-1})$, new samples are generated from a suitably designed proposal distribution, which may depend on the old state and the new measurements, i.e., $\mathbf{x}_t^{(n)} \sim q(\mathbf{x}_t \mid \mathbf{x}_{t-1}^{(n)}, \mathbf{y}_t)$, $n = 1 \dots N$. To maintain a consistent sample the new importance weights are set to

$$w_t^{(n)} \propto w_{t-1}^{(n)} \frac{p(\mathbf{y}_t \mid \mathbf{x}_t^{(n)}) p(\mathbf{x}_t^{(n)} \mid \mathbf{x}_{t-1}^{(n)})}{q(\mathbf{x}_t^{(n)} \mid \mathbf{x}_{t-1}^{(n)}, \mathbf{y}_t)}, \qquad \sum_{n=1}^{N} w_t^{(n)} = 1 \tag{3}$$

where the proportionality is up to a normalising constant. The new particle set $\{w_t^{(n)}, \mathbf{x}_t^{(n)}\}_{n=1}^{N}$ is then approximately distributed according to $p(\mathbf{x}_t \mid \mathbf{y}_{1:t})$. Approximations to the desired point estimates can then be obtained by Monte Carlo techniques. From time to time it is necessary to resample the particles to avoid degeneracy of the importance weights. The resampling procedure essentially multiplies particles with high importance weights, and discards those with low importance weights. A full discussion of degeneracy and resampling falls outside the scope of this work, but more detail can be found in [10].

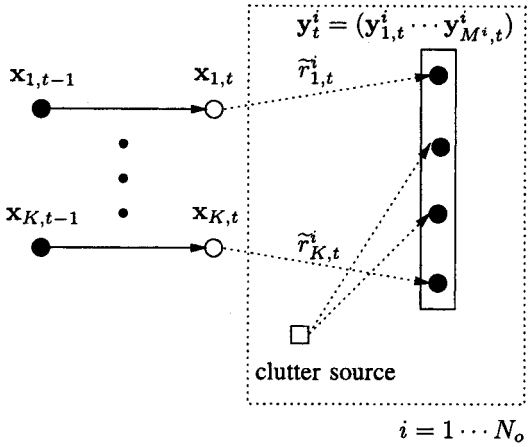


Fig. 1. Graphical model. Graphical model is given for single time step. Filled dots indicate known variables, open dots indicate unobserved random variables. Associations depicted in terms of target to measurement associations, and also unobserved.

III. MODEL DESCRIPTION

In this section we describe the elements of the multi-target tracking model. The construction of the state-space and the assumptions on the target dynamics are first treated in Section IIIA. Section IIIB then describes the measurement process and the data association problem, and formulates the likelihood conditional on a known association hypothesis. A prior for the association hypothesis is developed in Section IIIC. All the components of the model are elucidated by the graphical model in Fig. 1. Where appropriate we illustrate the various parts of the model with practical examples.

A. State-Space and Dynamics

We assume that the number of targets to be tracked K is fixed and known. Each target is

[1] The dynamic and likelihood models rely on the Markov assumptions

$$\mathbf{x}_t \perp \mathbf{y}_{1:t-1} \mid \mathbf{x}_{t-1}$$
$$\mathbf{y}_t \perp \mathbf{y}_{1:t-1} \mid \mathbf{x}_t \tag{2}$$

i.e., the current state is independent of all the previous measurements given the previous state, and the current measurements are independent of all the previous measurements given the current state.

parameterised by a state $\mathbf{x}_{k,t}$, $k = 1 \dots K$, which may differ in interpretation over the individual targets. The combined state is constructed as the concatenation of the individual target states, i.e., $\mathbf{x}_t = (\mathbf{x}_{1,t} \dots \mathbf{x}_{K,t})$. The individual targets are assumed to evolve independently according to Markovian dynamic models $p_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1})$, $k = 1 \dots K$. This implies that the dynamics for the combined state factorises over the individual targets, i.e.,

$$p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = \prod_{k=1}^{K} p_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}). \tag{4}$$

The tracking strategies we develop later can also be applied to targets whose motion is coupled, such as leader-follower behaviour and convoy movements. In such cases the targets that are correlated need to be treated jointly as a single "super target." For simplicity we do not explicitly consider this possibility here.

EXAMPLE 1 In the tracking application we consider later we are interested in tracking K slowly manoeuvring targets in the xy plane. We assume that each component of the position evolves independently according to a near constant velocity model of the form in [16]. The state of the kth target comprises its position and velocity in the xy plane, i.e., $\mathbf{x}_{k,t} = (x_{k,t}, \dot{x}_{k,t}, y_{k,t}, \dot{y}_{k,t})$. Assuming further a uniform discretisation with a sampling period of T seconds, the state evolution equation for the kth target becomes

$$\mathbf{x}_{k,t} = \mathbf{A}\mathbf{x}_{k,t-1} + \mathbf{v}_{k,t} \tag{5}$$

with

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{\text{cv}} & \mathbf{0}_{2\times 2} \\ \mathbf{0}_{2\times 2} & \mathbf{A}_{\text{cv}} \end{bmatrix}, \qquad \mathbf{A}_{\text{cv}} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \tag{6}$$

where $\mathbf{0}_{n\times m}$ denotes the $n \times m$ matrix of zeros. The state evolution noise $\mathbf{v}_{k,t}$ is assumed to be zero-mean Gaussian distributed with fixed and known covariance

$$\mathbf{\Sigma}_k = \begin{bmatrix} \sigma_{k,x}^2 \mathbf{\Sigma}_{\text{cv}} & \mathbf{0}_{2\times 2} \\ \mathbf{0}_{2\times 2} & \sigma_{k,y}^2 \mathbf{\Sigma}_{\text{cv}} \end{bmatrix}, \qquad \mathbf{\Sigma}_{\text{cv}} = \begin{bmatrix} T^3/3 & T^2/2 \\ T^2/2 & T \end{bmatrix}. \tag{7}$$

Note that under these assumptions the individual target dynamics are linear and Gaussian, and nonsingular.

B. Measurements, Data Association, and Likelihood

In the discussions that follow we suppress the time index t whenever there is no danger of ambiguity arising. Measurements for multi-target tracking are assumed to be available from N_o spatially distributed observers. We denote the observer locations by $\mathbf{p}_o^i$, $i = 1 \dots N_o$, and allow them to vary with time. At any particular time the combined set of measurements from all the observers will be denoted by $\mathbf{y} = (\mathbf{y}^1 \dots \mathbf{y}^{N_o})$, where $\mathbf{y}^i = (\mathbf{y}_1^i \dots \mathbf{y}_{M^i}^i)$ is the vector comprising the M^i measurements at the ith observer. Note that the number of measurements at each observer generally varies with time. The nature of the individual measurements $\mathbf{y}_j^i$ will depend on the characteristics of the sensors. Typically each measurement will correspond to an estimated line of sight from the observer location to the measurement source. Measurements do not only arise from the targets to be tracked. Additional clutter measurements may result due to multi-path effects, spurious objects, sensor errors, etc. We assume that each of the targets can generate at most one measurement per sensor at a particular time step, but may go undetected. We further assume that several or all of the measurements may be due to clutter.

To deal with the data association problem it is necessary to introduce a set of association variables. These can be specified either as measurement to target or target to measurement associations, with the former being more commonly used. Both formulations carry the same information, but are useful in different contexts. We consider both in what follows. We denote a measurement to target association ($\text{M} \rightarrow \text{T}$) hypothesis by $\lambda = (\lambda^1 \dots \lambda^{N_o})$, where $\lambda^i = (\mathbf{r}^i, M_C^i, M_T^i)$ is the measurement to target association hypothesis for the measurements at the ith observer, with M_C^i the number of clutter measurements, and M_T^i the number of target measurements. Note that $M^i = M_C^i + M_T^i$. The elements of the association vector $\mathbf{r}^i = (r_1^i \dots r_{M^i}^i)$ are given by

$$r_j^i = \begin{cases} 0 & \text{if measurement } j \text{ at observer } i \text{ is due to clutter} \\ k \in \{1 \dots K\} & \text{if measurement } j \text{ at observer } i \text{ stems from target } k. \end{cases} \tag{8}$$

In a similar fashion we denote a target to measurement association ($\text{T} \rightarrow \text{M}$) hypothesis by $\tilde{\lambda} = (\tilde{\lambda}^1 \dots \tilde{\lambda}^{N_o})$, where $\tilde{\lambda}^i = (\tilde{\mathbf{r}}^i, M_C^i, M_T^i)$ is the target to measurement hypothesis for the measurements at the ith observer. For this formulation the elements of the association vector $\tilde{\mathbf{r}}^i = (\tilde{r}_1^i \dots \tilde{r}_K^i)$ are given by

$$\tilde{r}_k^i = \begin{cases} 0 & \text{if target } k \text{ is undetected at observer } i \\ j \in \{1 \dots M^i\} & \text{if target } k \text{ generated measurement } j \text{ at observer } i. \end{cases} \tag{9}$$

As noted above these two representations are equivalent, and it is straightforward to convert from one to the other. More specifically, initialising the corresponding association vectors with zero, the

conversions for the ith observer are given by

$$\begin{aligned} &\mathrm{T}\to\mathrm{M} \quad \text{to} \quad \mathrm{M}\to\mathrm{T}: \quad r^i_{\tilde{r}^i_k} = k \qquad \text{if} \quad \tilde{r}^i_k \neq 0, \quad k = 1\ldots K \\ &\mathrm{M}\to\mathrm{T} \quad \text{to} \quad \mathrm{T}\to\mathrm{M}: \quad \tilde{r}^i_{r^i_j} = j \qquad \text{if} \quad r^i_j \neq 0, \quad j = 1\ldots M^i. \end{aligned} \tag{10}$$

Conditional on any formulation of the association hypothesis we assume the measurements at a particular observer to be independent of each other, and independent of those at the other observers. This leads to a factorised likelihood model which, conditional on the measurement to target association hypothesis, becomes

$$p(\mathbf{y} \mid \mathbf{x}, \lambda) = \prod_{i=1}^{N_o} \left[\prod_{j\in\mathcal{I}^i_0} p^i_C(\mathbf{y}^i_j) \cdot \prod_{j\in\mathcal{I}^i} p^i_T(\mathbf{y}^i_j \mid \mathbf{x}_{r^i_j}) \right] \tag{11}$$

where $\mathcal{I}^i_0 = \{j \in \{1\ldots M^i\} : r^i_j = 0\}$ and $\mathcal{I}^i = \{j \in \{1\ldots M^i\} : r^i_j \neq 0\}$[2] are, respectively, the subsets of measurement indices at the ith observer corresponding to clutter measurements and measurements from the targets to be tracked. In the above p^i_C denotes the clutter likelihood model for the ith observer, which is normally assumed to be uniform over the volume of the measurement space V^i. The likelihood for a measurement at the ith observer associated with a particular target, denoted by p^i_T, depends only on the state of the target with which it is associated. Under these assumptions the likelihood simplifies to

$$p(\mathbf{y} \mid \mathbf{x}, \lambda) = \prod_{i=1}^{N_o} \left[(V^i)^{-M^i_C} \prod_{j\in\mathcal{I}^i} p^i_T(\mathbf{y}^i_j \mid \mathbf{x}_{r^i_j}) \right]. \tag{12}$$

In a similar fashion the likelihood conditional on the target to measurement association hypothesis becomes

$$\begin{aligned} p(\mathbf{y} \mid \mathbf{x}, \tilde{\lambda}) &= \prod_{i=1}^{N_o} \left[\prod_{j\in\mathcal{I}^i_0} p^i_C(\mathbf{y}^i_j) \cdot \prod_{k=1}^{K} p^i(\mathbf{y}^i_{\tilde{r}^i_k} \mid \mathbf{x}_k) \right] \\ &= \prod_{i=1}^{N_o} \left[(V^i)^{-M^i_C} \prod_{k=1}^{K} p^i(\mathbf{y}^i_{\tilde{r}^i_k} \mid \mathbf{x}_k) \right] \end{aligned} \tag{13}$$

where, with a minor abuse of notation, the likelihood in the second product can be written as

$$p^i(\mathbf{y}^i_{\tilde{r}^i_k} \mid \mathbf{x}_k) = \begin{cases} 1 & \text{if} \quad \tilde{r}^i_k = 0 \\ p^i_T(\mathbf{y}^i_{\tilde{r}^i_k} \mid \mathbf{x}_k) & \text{otherwise.} \end{cases} \tag{14}$$

EXAMPLE 2 If the sensors yield line of sight measurements of the targets relative to the observers in the xy plane, the individual measurements at the ith observer can be written as $\mathbf{y}^i_j = (R^i_j, \theta^i_j)$, where R^i_j and θ^i_j are, respectively, the measured range and bearing from the observer to the source, with the bearing measured anti-clockwise from the x axis. If the range and bearing are assumed to be corrupted by independent Gaussian noise, the likelihood for the jth measurement, under the hypothesis that it is associated with the kth target, becomes

$$p^i_T(\mathbf{y}^i_j \mid \mathbf{x}_k) = \mathcal{N}(\mathbf{y}^i_j \mid \hat{\mathbf{y}}^i_k, \mathbf{\Sigma}^i_{\mathbf{y}}) \tag{15}$$

where $\mathbf{\Sigma}^i_{\mathbf{y}} = \mathrm{diag}(\sigma^2_{R^i}, \sigma^2_{\theta^i})$ is the fixed and known diagonal covariance with the individual noise variances. The components of the mean $\hat{\mathbf{y}}^i_k = (\hat{R}^i_k, \hat{\theta}^i_k) = \mathbf{g}(\mathbf{x}_k, \mathbf{p}^i_o)$ are given by

$$\begin{aligned} \hat{R}^i_k &= ((x_k - x^i_o)^2 + (y_k - y^i_o)^2)^{1/2} \\ \hat{\theta}^i_k &= \tan^{-1}\left(\frac{y_k - y^i_o}{x_k - x^i_o}\right) \end{aligned} \tag{16}$$

with $\mathbf{p}^i_o = (x^i_o, y^i_o)$. For this model the volume of the measurement space for the ith observer is $V^i = 2\pi R^i_{\max}$, where $R^i_{\max}$ is the maximum range of the sensor. This follows from the measurement independence assumption.

C. Association Prior

In most applications of practical interest the association hypothesis is unknown, and thus needs to be estimated alongside the other unknowns, or marginalised from the problem. To achieve this within a Bayesian framework it is necessary to define a prior distribution over the association hypothesis. We assume the prior for the association hypothesis to be independent of the state and past values of the association hypothesis. The prior we present here follows closely the one described in [41]. For the measurement to target association hypothesis we assume that the prior factorises over the observers, i.e.,

$$p(\lambda) = \prod_{i=1}^{N_o} p(\lambda^i). \tag{17}$$

For each of the observers the prior is further assumed to follow the hierarchical structure given by

$$p(\lambda^i) = p(\mathbf{r}^i \mid M^i_C, M^i_T) p(M^i_C) p(M^i_T) \tag{18}$$

with

$$\begin{aligned} p(\mathbf{r}^i \mid M^i_C, M^i_T) &= [N_{\lambda^i}(M^i_C, M^i_T)]^{-1} \\ p(M^i_C) &= (\lambda^i_C)^{M^i_C} \exp(-\lambda^i_C)/M^i_C! \\ p(M^i_T) &= \binom{K}{M^i_T} P_D^{M^i_T} (1 - P_D)^{K - M^i_T}. \end{aligned} \tag{19}$$

In the absence of measurements the prior for the association vector is assumed to be uniform over all the valid hypotheses. Given the number of target

[2] In terms of the target to measurement association hypothesis these sets can be denoted by $\mathcal{I}^i_0 = \{1\ldots M^i\} \setminus \{\tilde{r}^i_k : k = 1\ldots K\}$ and $\mathcal{I}^i = \{\tilde{r}^i_k : k = 1\ldots K\}$, respectively.

measurements, the number of valid hypotheses are given by

$$N_{\lambda^i}(M_C^i, M_T^i) = \binom{M^i}{M_T^i} \frac{K!}{(K - M_T^i)!} \tag{20}$$

and follows from the number of ways of choosing a subset of M_T^i elements from the available M^i measurements, multiplied by the number of possible associations between the M_T^i detections and the K targets. The number of clutter measurements is assumed to follow a Poisson distribution, with rate parameter λ_C^i,[3] which is assumed to be fixed and known. The binomial prior for the number of target measurements follows from summing over the $\binom{K}{M_T^i}$ possible ways to group M_T^i target detections among the K targets under the assumption that all the targets share the same fixed and known detection probability P_D. It is, of course, possible to associate different detection probabilities with each of the target-observer pairs, and have them vary with time, but we do not consider this possibility here.

The prior for the target to measurement association hypothesis follows a similar structure, with the only difference being that the number of valid hypotheses for a given number of target measurements is now given by

$$N_{\tilde{\lambda}^i}(M_C^i, M_T^i) = \binom{K}{M_T^i} \frac{M^i!}{(M^i - M_T^i)!}. \tag{21}$$

In the algorithms we develop later we make extensive use of this form of the association prior. For this form it is possible to obtain a factorisation over the individual target associations that takes the form

$$p(\tilde{\lambda}^i) = p(M_C^i) \prod_{k=1}^{K} p(\tilde{r}_k^i \mid \tilde{r}_{1:k-1}^i) \tag{22}$$

with

$$p(\tilde{r}_k^i = j \mid \tilde{r}_{1:k-1}^i) \propto \begin{cases} 1 - P_D & \text{if} \quad j = 0 \\ 0 & \text{if} \quad j > 0 \quad \text{and} \quad j \in \{\tilde{r}_1^i \ldots \tilde{r}_{k-1}^i\} \\ \dfrac{P_D}{M_k^i} & \text{otherwise,} \end{cases} \tag{23}$$

where $M_k^i = M^i - |\{l : \tilde{r}_l^i \neq 0, l = 1 \ldots k-1\}|$ is the number of unassigned measurements, taking into account the assignments of the previous $k-1$ associations. Note that this sequential factorisation can be performed over any permutation of the target ordering. Note further that the prior for the number of target detections is implicitly captured by the factorisation of the association vector, and hence disappears from the expression for the prior. This factorisation will aid in the design of efficient sampling strategies to combat the curse of dimensionality with an increase in the number of targets.

[3] It is common practice to define the rate parameter in terms of the spatial density of the clutter μ^i, i.e., $\lambda_C^i = \mu^i \tilde{V}^i$. Here $\tilde{V}^i$ is the physical volume of the space observed by the sensor, to be contrasted with the volume of the measurement space V^i.

IV. MONTE CARLO JPDAF

The JPDAF is probably the most widely applied and successful strategy for multi-target tracking under data association uncertainty. The original formulation of the JPDAF in [16] and [17] assumes linear and Gaussian models. For models with weak nonlinearities the EKF can be applied to linearise the system. Subsequent research has led to many useful extensions to the standard JPDAF to address some of its shortcomings and make it more generally applicable. For example, methods to maintain a mixture of Gaussians for each target state, instead of a single Gaussian, are described in [18] and [19].

Here we present a Monte Carlo implementation of the general JPDAF framework, applicable to general nonlinear and non-Gaussian models. This strategy, which we refer to as the MC-JPDAF, is a generalisation of the method proposed in [1] and [2] to multiple observers and arbitrary proposal distributions. We first outline the general JPDAF framework in Section IVA, and then show in Section IVB how this framework can be implemented using Monte Carlo techniques, so that it applies to general nonlinear and non-Gaussian models. The JPDAF requires a gating procedure to keep the number of valid association hypotheses to a reasonable level. In Section IVC we describe a gating procedure that is applicable within the context of a Monte Carlo implementation of the JPDAF.

A. JPDAF Framework

Instead of maintaining the filtering distribution for the joint state $p(\mathbf{x}_t \mid \mathbf{y}_{1:t})$ the JPDAF effectively combats the curse of dimensionality by recursively updating the marginal filtering distributions for each of the targets $p_k(\mathbf{x}_{k,t} \mid \mathbf{y}_{1:t})$, $k = 1 \ldots K$, through the Bayesian sequential estimation recursions in (1). The prediction step proceeds independently for each target as

$$p_k(\mathbf{x}_{k,t} \mid \mathbf{y}_{1:t-1}) = \int p_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}) p_k(\mathbf{x}_{k,t-1} \mid \mathbf{y}_{1:t-1}) d\mathbf{x}_{k,t-1}. \tag{24}$$

Due to the data association uncertainty the filtering step cannot be performed independently for the individual targets. The JPDAF gets around this

difficulty by performing a soft assignment of targets to measurements according to the corresponding posterior probabilities of these marginal associations. More specifically, it achieves this by defining the likelihood for the kth target as

$$p_k(\mathbf{y}_t \mid \mathbf{x}_{k,t}) = \prod_{i=1}^{N_o} \left[\beta_{0k}^i + \sum_{j=1}^{M^i} \beta_{jk}^i p_T^i(\mathbf{y}_{j,t}^i \mid \mathbf{x}_{k,t}) \right] \quad (25)$$

where $\beta_{jk}^i = p(\tilde{r}_{k,t}^i = j \mid \mathbf{y}_{1:t})$, $j = 1 \ldots M^i$, is the posterior probability that the kth target is associated with the jth measurement, with β_{0k}^i the posterior probability that the kth target is undetected. The likelihood is assumed to be independent over the observers. The component of the likelihood for each observer is a mixture, with one mixture component for each possible target to measurement association, and the mixture weights equal to the posterior probabilities of the corresponding marginal associations. With the definition of the likelihood above the filtering step proceeds in the normal way specified by (1), i.e.,

$$p_k(\mathbf{x}_{k,t} \mid \mathbf{y}_{1:t}) \propto p_k(\mathbf{y}_t \mid \mathbf{x}_{k,t}) p_k(\mathbf{x}_{k,t} \mid \mathbf{y}_{1:t-1}). \quad (26)$$

Thus with the definitions for the one step ahead prediction distribution in (24) and the filtering distribution in (26) the JPDAF fits within the Bayesian sequential estimation framework of (1). All that remains is the computation of the posterior probabilities of the marginal associations β_{jk}^i, where $i = 1 \ldots N_o$ ranges over the observers, $j = 0 \ldots M^i$ ranges over the measurements, with 0 signifying that the target in question is not associated with any of the measurements, and $k = 1 \ldots K$ ranges over the targets. These marginal probabilities can be computed by summing over the corresponding joint association probabilities, i.e.,

$$\beta_{jk}^i = p(\tilde{r}_{k,t}^i = j \mid \mathbf{y}_{1:t}) = \sum_{\{\tilde{\lambda}_t^i \in \tilde{\Lambda}_t^i : \tilde{r}_{k,t}^i = j\}} p(\tilde{\lambda}_t^i \mid \mathbf{y}_{1:t}) \quad (27)$$

where $\tilde{\Lambda}_t^i$ is the set of all valid joint target to measurement association hypotheses for the data at the ith observer. Thus the posterior probability for a particular target to measurement association is obtained by summing over all the joint association hypotheses in which this marginal association occurs. Under the assumptions discussed below the posterior probability for the joint association hypothesis can be expressed as

$$p(\tilde{\lambda}_t^i \mid \mathbf{y}_{1:t}) \propto p(\tilde{\lambda}_t^i)(V^i)^{-M_C^i} \prod_{j \in \mathcal{I}^i} p_{r_{j,t}^i}(\mathbf{y}_{j,t}^i \mid \mathbf{y}_{1:t-1}) \quad (28)$$

where $p(\tilde{\lambda}_t^i)$ is the joint association prior developed in Section IIIC, and $p_k(\mathbf{y}_{j,t}^i \mid \mathbf{y}_{1:t-1})$ is the predictive likelihood for the jth measurement at the ith observer using the information from the kth target, given in the standard way by

$$p_k(\mathbf{y}_{j,t}^i \mid \mathbf{y}_{1:t-1}) = \int p_T^i(\mathbf{y}_{j,t}^i \mid \mathbf{x}_{k,t}) p_k(\mathbf{x}_{k,t} \mid \mathbf{y}_{1:t-1}) d\mathbf{x}_{k,t}. \quad (29)$$

To obtain the expression for the joint association posterior probability in (28) it was necessary to assume that the predictive likelihood is independent over the observers, as well as over the individual measurements at each of the observers. In the likelihood the conditioning was also changed from the target to measurement associations to the measurement to target associations. This is valid since both representations carry the same information, and are related through a deterministic one-to-one mapping.

The original formulation of the JPDAF in [16] and [17] assumes linear and Gaussian forms for the dynamic and likelihood models, and a Gaussian approximation for the filtering distribution. Under these assumptions Kalman filter updates are obtained for the one step ahead prediction distribution in (24) and the predictive likelihood in (29). The mixture likelihood in (25) is collapsed into a single Gaussian, so that a Kalman filter update is also obtained for the filtering distribution in (26). In the next section we show how the general JPDAF framework can be implemented using Monte Carlo techniques, making it applicable to general nonlinear and non-Gaussian models.

B. Monte Carlo Implementation

The Monte Carlo implementation of the JPDAF presented in this section is a generalisation of the strategy proposed in [1] and [2] to multiple observers and arbitrary proposal distributions. It aims to represent the marginal filtering distributions for each of the targets with Monte Carlo samples, or particles, instead of a Gaussian, as is the case for the standard JPDAF. More specifically, for the kth target, assume that a set of samples $\{w_{k,t-1}^{(n)}, \mathbf{x}_{k,t-1}^{(n)}\}_{n=1}^N$ is available, approximately distributed according to the marginal filtering distribution at the previous time step $p_k(\mathbf{x}_{k,t-1} \mid \mathbf{y}_{1:t-1})$. At the current time step new samples for the target state are generated from a suitably constructed proposal distribution, which may depend on the old state and the new measurements, i.e.,

$$\mathbf{x}_{k,t}^{(n)} \sim q_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}^{(n)}, \mathbf{y}_t), \qquad n = 1 \ldots N. \quad (30)$$

We are not specific about the form of the state proposal distribution here, but will delay the presentation of particular forms for this distribution until Section VI. Using these Monte Carlo samples the predictive likelihoods in (29) can straightforwardly be approximated as

$$p_k(\mathbf{y}_{j,t}^i \mid \mathbf{y}_{1:t-1}) \approx \sum_{n=1}^{N} \alpha_{k,t}^{(n)} p_T^i(\mathbf{y}_{j,t}^i \mid \mathbf{x}_{k,t}^{(n)}) \quad (31)$$

where the predictive weights are given by

$$\alpha_{k,t}^{(n)} \propto w_{k,t-1}^{(n)} \frac{p_k(\mathbf{x}_{k,t}^{(n)} \mid \mathbf{x}_{k,t-1}^{(n)})}{q_k(\mathbf{x}_{k,t}^{(n)} \mid \mathbf{x}_{k,t-1}^{(n)}, \mathbf{y}_t)}, \qquad \sum_{n=1}^{N} \alpha_{k,t}^{(n)} = 1. \tag{32}$$

This approximation can now straightforwardly be substituted into (28) to obtain approximations for the joint association posterior probabilities, from which approximations for the marginal target to measurement association posterior probabilities can be computed according to (27). These approximations can, in turn, be used in (25) to approximate the target likelihood. Finally, setting the new importance weights to

$$w_{k,t}^{(n)} \propto w_{k,t-1}^{(n)} \frac{p_k(\mathbf{y}_t \mid \mathbf{x}_{k,t}^{(n)}) p_k(\mathbf{x}_{k,t}^{(n)} \mid \mathbf{x}_{k,t-1}^{(n)})}{q_k(\mathbf{x}_{k,t}^{(n)} \mid \mathbf{x}_{k,t-1}^{(n)}, \mathbf{y}_t)}, \qquad \sum_{n=1}^{N} w_{k,t}^{(n)} = 1 \tag{33}$$

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

We conclude this section by presenting a summary of the MC-JPDAF algorithm. Assuming that the sample sets $\{w_{k,t-1}^{(n)}, \mathbf{x}_{k,t-1}^{(n)}\}_{n=1}^{N}$, $k = 1 \dots K$, are approximately distributed according to the corresponding marginal filtering distributions at the previous time step $p_k(\mathbf{x}_{k,t-1} \mid \mathbf{y}_{1:t-1})$, $k = 1 \dots K$, the algorithm proceeds as follows at the current time step.

ALGORITHM 1 MC-JPDAF

• For $k = 1 \dots K$, $n = 1 \dots N$, generate new samples for the target states $\mathbf{x}_{k,t}^{(n)} \sim q_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}^{(n)}, \mathbf{y}_t)$.

• For $k = 1 \dots K$, $n = 1 \dots N$, compute and normalise the predictive weights

$$\alpha_{k,t}^{(n)} \propto w_{k,t-1}^{(n)} \frac{p_k(\mathbf{x}_{k,t}^{(n)} \mid \mathbf{x}_{k,t-1}^{(n)})}{q_k(\mathbf{x}_{k,t}^{(n)} \mid \mathbf{x}_{k,t-1}^{(n)}, \mathbf{y}_t)}, \qquad \sum_{n=1}^{N} \alpha_{k,t}^{(n)} = 1. \tag{34}$$

• For $k = 1 \dots K$, $i = 1 \dots N_o$, $j = 1 \dots M^i$, compute the Monte Carlo approximation for the predictive likelihood

$$p_k(\mathbf{y}_{j,t}^i \mid \mathbf{y}_{1:t-1}) \approx \sum_{n=1}^{N} \alpha_{k,t}^{(n)} p_T^i(\mathbf{y}_{j,t}^i \mid \mathbf{x}_{k,t}^{(n)}). \tag{35}$$

• For $i = 1 \dots N_o$, enumerate all the valid joint target to measurement association hypotheses at the ith observer to form the set $\tilde{\Lambda}_t^i$.

• For $i = 1 \dots N_o$, $\tilde{\lambda}_t^i \in \tilde{\Lambda}_t^i$, compute the joint association posterior probability

$$p(\tilde{\lambda}_t^i \mid \mathbf{y}_{1:t}) \propto p(\tilde{\lambda}_t^i)(V^i)^{-M_C^i} \prod_{j \in \mathcal{T}^i} p_{r_{j,t}^i}(\mathbf{y}_{j,t}^i \mid \mathbf{y}_{1:t-1}). \tag{36}$$

• For $k = 1 \dots K$, $i = 1 \dots N_o$, $j = 0 \dots M^i$, compute the marginal association posterior probability

$$\beta_{jk}^i = \sum_{\{\tilde{\lambda}_t^i \in \tilde{\Lambda}_t^i : \tilde{r}_{k,t}^i = j\}} p(\tilde{\lambda}_t^i \mid \mathbf{y}_{1:t}). \tag{37}$$

• For $k = 1 \dots K$, $n = 1 \dots N$, compute the target likelihood

$$p_k(\mathbf{y}_t \mid \mathbf{x}_{k,t}^{(n)}) = \prod_{i=1}^{N_o} \left[\beta_{0k}^i + \sum_{j=1}^{M^i} \beta_{jk}^i p_T^i(\mathbf{y}_{j,t}^i \mid \mathbf{x}_{k,t}^{(n)}) \right]. \tag{38}$$

• For $k = 1 \dots K$, $n = 1 \dots N$, compute and normalise the particle weights

$$w_{k,t}^{(n)} \propto w_{k,t-1}^{(n)} \frac{p_k(\mathbf{y}_t \mid \mathbf{x}_{k,t}^{(n)}) p_k(\mathbf{x}_{k,t}^{(n)} \mid \mathbf{x}_{k,t-1}^{(n)})}{q_k(\mathbf{x}_{k,t}^{(n)} \mid \mathbf{x}_{k,t-1}^{(n)}, \mathbf{y}_t)}, \qquad \sum_{n=1}^{N} w_{k,t}^{(n)} = 1. \tag{39}$$

• For $k = 1 \dots K$, if resampling is required then for $n = 1 \dots N$, sample an index $m(n) \sim \{w_{k,t}^{(l)}\}_{l=1}^{N}$ and replace $\{w_{k,t}^{(n)}, \mathbf{x}_{k,t}^{(n)}\} \leftarrow \{N^{-1}, \mathbf{x}_{k,t}^{m(n)}\}$.

The resulting sample sets $\{w_{k,t}^{(n)}, \mathbf{x}_{k,t}^{(n)}\}_{n=1}^{N}$, $k = 1 \dots K$, are then approximately distributed according to the corresponding marginal filtering distributions at the current time step $p_k(\mathbf{x}_{k,t} \mid \mathbf{y}_{1:t})$, $k = 1 \dots K$. Note that the resampling step can be performed independently for each of the targets. Resampling is necessary to avoid degeneracy of the particle sets, and is normally invoked once an estimate of the effective sample size, which acts as a measure of degeneracy, falls below a predefined threshold. A more complete discussion about degeneracy and resampling can be found in [10].

C. Gating

The biggest drawback of the JPDAF is its computational complexity. Recall that the computation of the marginal association posterior probabilities in (27) requires enumeration over all the valid joint target to measurement associations. For a particular observer the total number of such associations is given by[4]

$$N_{\tilde{\lambda}} = \sum_{M_T=0}^{\min(K,M)} N_{\tilde{\lambda}}(M_C, M_T) \tag{40}$$

where the number of hypotheses for a given number of target detections and clutter measurements $N_{\tilde{\lambda}}(M_C, M_T)$ follows from the expression in (21). The

[4]The issues surrounding gating applies independently to each of the observers. We thus focus on a single observer, and drop the observer index in the discussion for notational clarity. For the same reason we also suppress the time index.

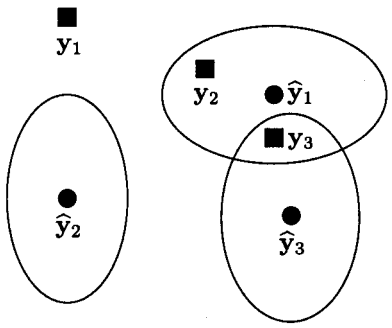


Fig. 2. Measurement gating. Targets mapped into measurement space (circles) and their validation regions (ellipses). Only measurements (squares) that fall inside validation region for a particular target are candidates to be associated with that target.

number of valid hypotheses increases exponentially with an increase in the number of measurements and targets, rendering exhaustive enumeration infeasible for even moderate values for these quantities. Thus if the JPDAF is to remain a practical alternative for solving the multi-target tracking problem methods are required to reduce the number of hypotheses to a feasible level.

Gating [16] is probably one of the most straightforward and effective methods to achieve this reduction. For each target a validation region is constructed from the available information, and only measurements that fall within the target validation region are considered as possible candidates to be associated with the particular target. This procedure is graphically illustrated in Fig. 2.

We now present a gating procedure that can be applied within the context of the MC-JPDAF. The Monte Carlo approximation of the predictive likelihood in (31) can be written as

$$p_k(\mathbf{y} \mid \mathbf{y}_{1:t-1}) \approx \sum_{n=1}^{N} \alpha_k^{(n)} \mathcal{N}(\mathbf{y} \mid \hat{\mathbf{y}}_k^{(n)}, \boldsymbol{\Sigma}_{\mathbf{y}}) \tag{41}$$

where $\hat{\mathbf{y}}_k = \mathbf{g}(\mathbf{x}_k, \mathbf{p}_o)$ is the mapping of the kth target state into the measurement space, as illustrated in Example 2. This Gaussian mixture can be straightforwardly approximated by a single Gaussian with mean and covariance given by

$$\begin{aligned} \boldsymbol{\mu}_{\hat{\mathbf{y}}_k} &= \sum_{n=1}^{N} \alpha_k^{(n)} \mathbf{g}(\mathbf{x}_k^{(n)}, \mathbf{p}_o) \\ \boldsymbol{\Sigma}_{\hat{\mathbf{y}}_k} &= \boldsymbol{\Sigma}_{\mathbf{y}} + \sum_{n=1}^{N} \alpha_k^{(n)} [\mathbf{g}(\mathbf{x}_k^{(n)}, \mathbf{p}_o) - \boldsymbol{\mu}_{\hat{\mathbf{y}}_k}][\mathbf{g}(\mathbf{x}_k^{(n)}, \mathbf{p}_o) - \boldsymbol{\mu}_{\hat{\mathbf{y}}_k}]^{\mathrm{T}}. \end{aligned} \tag{42}$$

Using these statistics a set of validated measurements for the kth target can be obtained as

$$\mathcal{Y}_k = \{\mathbf{y}_j : d_k^2(\mathbf{y}_j) \le \varepsilon\} \tag{43}$$

TABLE I
Viable Joint Associations for Configuration in Fig. 2

$\tilde{r}_1$	$\tilde{r}_2$	$\tilde{r}_3$
0	0	0
0	0	3
2	0	0
2	0	3
3	0	0

where $d_k^2(\mathbf{y}_j)$ is the squared distance based on the measurement innovations, given by

$$d_k^2(\mathbf{y}_j) = (\mathbf{y}_j - \boldsymbol{\mu}_{\hat{\mathbf{y}}_k})^{\mathrm{T}} \boldsymbol{\Sigma}_{\hat{\mathbf{y}}_k}^{-1} (\mathbf{y}_j - \boldsymbol{\mu}_{\hat{\mathbf{y}}_k}) \tag{44}$$

and ε is a parameter determining the size of the validation region. The validation region is an ellipsoid that contains a given probability mass under the Gaussian assumption. The value of the parameter ε is related to the specified value of the probability mass to be included in the validation region. It can be obtained using the fact that d_k^2 is approximately chi-squared distributed with number of degrees of freedom equal to the dimension of $\mathbf{y}_j$. The set of valid target to measurement associations for the kth target follows straightforwardly as

$$\tilde{r}_k \in \tilde{\mathcal{R}}_k = \{j : d_k^2(\mathbf{y}_j) \le \varepsilon\} \cup \{0\}. \tag{45}$$

Note that we always allow each of the targets to be undetected to take account of the possibility that any or all of the measurements within the target validation region may be due to clutter. As an example the valid association sets for the configuration in Fig. 2 are $\tilde{\mathcal{R}}_1 = \{0,2,3\}$, $\tilde{\mathcal{R}}_2 = \{0\}$ and $\tilde{\mathcal{R}}_3 = \{0,3\}$. The set of viable joint associations $\tilde{\Lambda}$ can now be constructed by enumerating all the valid combinations of the elements in the marginal sets $\tilde{\mathcal{R}}_k$, $k = 1 \ldots K$. The number of hypotheses obtained in this manner will typically be substantially smaller than the number obtained by an exhaustive enumeration. To complete our example the set of viable joint associations for the configuration in Fig. 2 is enumerated in Table I. There are only 5 such joint associations, whereas an exhaustive enumeration would have resulted in 34 joint associations, almost an order of magnitude more.

REMARK 1 The validation region for the kth target is given by $\mathcal{V}_k = \{\mathbf{y} : d_k^2(\mathbf{y}) \le \varepsilon\}$. The probability that a target measurement falls inside the validation region P_G can be obtained by integrating the predictive likelihood over the validation region. This integration is intractable, but a Monte Carlo approximation follows as

$$P_G = \int_{\mathcal{V}_k} p_k(\mathbf{y} \mid \mathbf{y}_{1:t-1}) d\mathbf{y} \approx \sum_{n=1}^{N} \alpha_k^{(n)} \int_{\mathcal{V}_k} \mathcal{N}(\mathbf{y} \mid \hat{\mathbf{y}}_k^{(n)}, \boldsymbol{\Sigma}_{\mathbf{y}}) d\mathbf{y}. \tag{46}$$

The validation region is generally not aligned with the likelihood covariance, so that the Gaussian integral above also becomes intractable. However, a Monte Carlo approximation can be straightforwardly obtained by generating samples from the Gaussian, and computing the proportion of the samples that fall inside the validation region, i.e.,

$$\int_{\mathcal{V}_k} \mathcal{N}(\mathbf{y} \mid \hat{\mathbf{y}}_k^{(n)}, \boldsymbol{\Sigma}_{\mathbf{y}}) d\mathbf{y} \approx L^{-1} \sum_{l=1}^{L} \mathbb{I}_{\mathcal{V}_k}(\mathbf{y}^{(l)} + \hat{\mathbf{y}}_k^{(n)}), \quad \{\mathbf{y}^{(l)}\}_{l=1}^{L} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\mathbf{y}}) \tag{47}$$

where $\mathbb{I}_{\mathcal{A}}(\cdot)$ denotes the indicator function for the set $\mathcal{A}$. Note that since the likelihood covariance is fixed only a single set of samples is required to compute the Gaussian integrals for all the terms in (46). Subsequent to gating we need to replace the target detection probability P_D in the association prior with $P_D P_G$, and restrict the predictive likelihood in (36) to the validation region, i.e., normalise it with P_G. In all our experiments we found that P_G was either unity, or very close to it, so we ignored it, without detrimental effects to the results.

V. MULTI-TARGET PARTICLE FILTERS

The JPDAF, even in its general form, suffers from a number of shortcomings. It is only an approximate method in that a number of independence assumptions are made to perform both the filtering operations and the computation of the association probabilities. Furthermore, any practical implementation requires a gating procedure to keep the number of association hypotheses to a feasible level. In this section we propose particle filtering strategies to solve the multi-target tracking problem. The first strategy samples the individual targets sequentially by utilising a factorisation of the importance weights. We refer to this algorithm as the SSPF. The second strategy assumes the associations to be independent over the individual targets. This is similar to the approximation made in the PMHT, and implies that measurements can be assigned to more than one target. This assumption effectively removes all dependencies between the individual targets, leading to an efficient component-wise sampling strategy to construct new particles. We refer to this approach as the Independent partition particle filter (IPPF). As opposed to the JPDAF, neither approach requires a gating procedure, but captures the notion of a soft-gating of the measurements by an efficiently designed proposal distribution.

The particle filtering algorithms differ from the MC-JPDAF in the manner in which they deal with the data association uncertainty and their strategy to combat the curse of dimensionality. The MC-JPDAF takes care of the association uncertainty by combining all of the feasible association hypotheses according to their corresponding posterior probabilities, which are in turn computed using the Monte Carlo samples. On the other hand, the particle filtering strategies augment the multi-target state with the unknown association hypothesis. The association uncertainty is then represented by Monte Carlo samples that are generated from an efficient proposal distribution based on the notion of a soft-gating of the measurements. The MC-JPDAF effectively avoids the curse of dimensionality by maintaining the marginal filtering distributions for each of the targets. The particle filtering strategies, on the other hand, exploit a factorisation of the importance weights to decompose the difficult joint estimation problem into a number of easier estimation problems, each defined in the state-space for a single target.

In what follows we first outline the standard particle filter architecture, adapted for multi-target tracking in Section VA. This strategy suffers greatly from the curse of dimensionality. Sections VB and VC then formulate the details of the SSPF and the IPPF, respectively, as alternatives to the standard particle filter for more efficient multi-target tracking.

A. Standard Particle Filter

Direct estimation of the filtering distribution of the joint target state $p(\mathbf{x}_t \mid \mathbf{y}_{1:t})$, with $\mathbf{x}_t = (\mathbf{x}_{1,t} \dots \mathbf{x}_{K,t})$, is difficult due to the unknown associations. To overcome this difficulty we augment the joint target state with the unknown associations, and attempt to estimate the joint filtering distribution $p(\mathbf{x}_t, \tilde{\lambda}_t \mid \mathbf{y}_{1:t})$ recursively through time using particle techniques. For the purpose of the particle filtering strategies introduced in the following sections we choose to work with the target to measurement associations. Using Bayes' rule the joint filtering distribution can be expressed as

$$\begin{aligned} p(\mathbf{x}_t, \tilde{\lambda}_t \mid \mathbf{y}_{1:t}) &\propto p(\tilde{\lambda}_t) p(\mathbf{y}_t \mid \mathbf{x}_t, \tilde{\lambda}_t) \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) \\ &\quad \times \sum_{\tilde{\lambda}_{t-1}} p(\mathbf{x}_{t-1}, \tilde{\lambda}_{t-1} \mid \mathbf{y}_{1:t-1}) d\mathbf{x}_{t-1} \\ &= p(\tilde{\lambda}_t) p(\mathbf{y}_t \mid \mathbf{x}_t, \tilde{\lambda}_t) \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) p(\mathbf{x}_{t-1} \mid \mathbf{y}_{1:t-1}) d\mathbf{x}_{t-1} \end{aligned} \tag{48}$$

where $p(\tilde{\lambda}_t)$ is the target to measurement association prior in (22), $p(\mathbf{y}_t \mid \mathbf{x}_t, \tilde{\lambda}_t)$ is the conditional likelihood in (13), and $p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$ is the joint state dynamics in (4). As for the standard particle filter, assume that a set of samples $\{w_{t-1}^{(n)}, \mathbf{x}_{t-1}^{(n)}, \tilde{\lambda}_{t-1}^{(n)}\}_{n=1}^{N}$ is available, approximately distributed according to the filtering distribution at the previous time step

$p(\mathbf{x}_{t-1}, \tilde{\lambda}_{t-1} \mid \mathbf{y}_{1:t-1})$. The samples for the joint state alone $\{w_{t-1}^{(n)}, \mathbf{x}_{t-1}^{(n)}\}_{n=1}^{N}$ are then approximately distributed according to the marginal filtering distribution at the previous time step $p(\mathbf{x}_{t-1} \mid \mathbf{y}_{1:t-1})$. These samples are sufficient to obtain a Monte Carlo approximation of the integral in (48), so that old samples for the associations can be discarded. This is due to the fact that the association hypotheses are assumed to be temporally independent. At the current time step new multi-target states and association hypotheses are jointly proposed from some appropriately defined proposal distribution, i.e.,

$$(\tilde{\lambda}_t^{(n)}, \mathbf{x}_t^{(n)}) \sim q(\tilde{\lambda}_t, \mathbf{x}_t \mid \mathbf{x}_{t-1}^{(n)}, \mathbf{y}_t), \qquad n = 1 \ldots N \quad (49)$$

where we define the joint proposal so that it factorises as

$$q(\tilde{\lambda}_t, \mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{y}_t) = q(\tilde{\lambda}_t \mid \mathbf{x}_t, \mathbf{y}_t) q(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{y}_t). \quad (50)$$

Note that the proposal for the association hypothesis depends only on information at the current time step. It is straightforward to show that setting the new importance weights to

$$w_t^{(n)} \propto w_{t-1}^{(n)} \frac{p(\tilde{\lambda}_t^{(n)}) p(\mathbf{y}_t \mid \mathbf{x}_t^{(n)}, \tilde{\lambda}_t^{(n)}) p(\mathbf{x}_t^{(n)} \mid \mathbf{x}_{t-1}^{(n)})}{q(\tilde{\lambda}_t^{(n)} \mid \mathbf{x}_t^{(n)}, \mathbf{y}_t) q(\mathbf{x}_t^{(n)} \mid \mathbf{x}_{t-1}^{(n)}, \mathbf{y}_t)}, \qquad \sum_{n=1}^{N} w_t^{(n)} = 1 \quad (51)$$

leads to the sample set $\{w_t^{(n)}, \mathbf{x}_t^{(n)}, \tilde{\lambda}_t^{(n)}\}_{n=1}^{N}$ being approximately distributed according to the joint filtering distribution at the current time step $p(\mathbf{x}_t, \tilde{\lambda}_t \mid \mathbf{y}_{1:t})$.

However, such a straightforward implementation of the particle filter suffers greatly from the curse of dimensionality. As the number of targets increases an exponentially increasing number of particles is required to maintain the same estimation accuracy. In practice, after the proposal step, it commonly occurs that the resulting joint particles have good estimates for some target states, and bad estimates for other target states. In the computation of the importance weights, however, the entire particle is penalised for the component targets with bad estimates, so that even components with good estimates are destroyed during the resampling stage. This leads to a rapid depletion of the Monte Carlo representation. The following two sections develop particle filtering strategies to combat this problem.

B. Sequential Sampling Particle Filter

If the association hypothesis were known the filtering distribution would factorise completely over the individual targets. Each of the targets could then be treated independently, thus defeating the curse of dimensionality. For an unknown association hypothesis a complete factorisation of the filtering distribution is not possible. It is, however, possible to construct a proposal for the associations that factorises sequentially over the individual target associations. This facilitates a strategy where the targets and their associations can be sampled sequentially, conditionally on each other, in much the same way as the standard particle filter samples states over time. We first describe the architecture for such a particle filter, which we term the SSPF, in Section VB1, before presenting the details for the association proposal in Section VB2.

1) *SSPF Architecture*: As was the case for the MC-JPDAF we assume the proposal for the joint target to factorise over the individual targets, i.e.,

$$q(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{y}_t) = \prod_{k=1}^{K} q_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}, \mathbf{y}_t). \quad (52)$$

Specific forms for the state proposal for the individual targets are presented in Section VI. In turn we assume the proposal for the association hypothesis to factorise over the individual observers, i.e.,

$$q(\tilde{\lambda}_t \mid \mathbf{x}_t, \mathbf{y}_t) = \prod_{i=1}^{N_o} q(\tilde{\lambda}_t^i \mid \mathbf{x}_t, \mathbf{y}_t^i). \quad (53)$$

For each individual observer we define the proposal in terms of the target to measurement association vector, i.e., $q(\tilde{\lambda}_t^i \mid \mathbf{x}_t, \mathbf{y}_t^i) = q(\tilde{\mathbf{r}}_t^i \mid \mathbf{x}_t, \mathbf{y}_t^i)$, with the proposals for the number of clutter measurements M_C^i and target detections M_T^i being implicit. Thus once the association vector has been generated from the proposal, the number of clutter measurements and target detections are deterministically calculated. However, we define the proposal for the association vector such that the resulting number of clutter measurements and target detections approximately follow the corresponding prior models in (19). For the time being we assume that the proposal for the association vector takes a form that factorises sequentially over the individual target associations, i.e.,

$$q(\tilde{\mathbf{r}}_t^i \mid \mathbf{x}_t, \mathbf{y}_t^i) = \prod_{k=1}^{K} q(\tilde{r}_{k,t}^i \mid \tilde{r}_{1:k-1,t}^i, \mathbf{x}_{k,t}, \mathbf{y}_t^i). \quad (54)$$

Note that the sequential factorisation can be performed over any permutation of the individual target associations. In practice we choose the order randomly, as discussed below. However, for notational clarity we retain the form above in the remainder of the discussion. We delay the development of the proposal for the individual target associations in the factorisation above until Section VB2, and turn now to the problem of computing the importance weights. Substituting the expressions for the conditional likelihood and joint state dynamics in (13) and (4), respectively, the factorised form for the association prior in (22), and the factorised forms for the state and association proposals in (52) and (53), respectively,

into (51) leads to an expression for the importance weight given by

$$w_t \propto w_{t-1}\left[\prod_{i=1}^{N_o}(V^i)^{-M_C^i}p(M_C^i)\right]\cdot\prod_{k=1}^{K}w_{k,t} \tag{55}$$

where the weight for the kth target is given by

$$w_{k,t} \propto \frac{p_k(\mathbf{x}_{k,t}\mid\mathbf{x}_{k,t-1})}{q_k(\mathbf{x}_{k,t}\mid\mathbf{x}_{k,t-1},\mathbf{y}_t)}\cdot\prod_{i=1}^{N_o}\frac{p^i(\mathbf{y}^i_{\tilde{r}^i_{k,t},t}\mid\mathbf{x}_{k,t})p(\tilde{r}^i_{k,t}\mid\tilde{r}^i_{1:k-1,t})}{q(\tilde{r}^i_{k,t}\mid\tilde{r}^i_{1:k-1,t},\mathbf{x}_{k,t},\mathbf{y}^i_t)}. \tag{56}$$

Note that the weight for the kth target $w_{k,t}$ depends on those for the targets earlier in the sequence only through the associations. This factorisation of the importance weight suggests a sampling procedure where the joint target state and association vector are constructed in a sequential fashion in much the same way as the standard particle filter samples states over time. The importance weight for the particle can be updated cumulatively during this construction, facilitating resampling steps should the variance of the cumulative weights become too high (i.e., the Monte Carlo representation becomes degenerate). Because of the resampling procedures the target sampled first in the sequence will have the most depleted representation, with the diversity increasing towards the target sampled last in the sequence. To eliminate this problem the procedure can be repeated a number of times for different orderings of the targets, with the final Monte Carlo representation obtained by combining the samples obtained from the individual runs.

The SSPF algorithm is summarised below. For the sake of clarity the summary is given for a single forward run of the algorithm. In practice the forward run should be repeated a number of times for different random permutations of the target ordering. The final result can then be obtained by treating the resulting discrete distributions as components of a mixture distribution, and sampling the required number of samples from this mixture. Assuming that the sample set $\{w^{(n)}_{t-1},\mathbf{x}^{(n)}_{t-1}\}^N_{n-1}$ is approximately distributed according to the marginal filtering distribution at the previous time step $p(\mathbf{x}_{t-1}\mid\mathbf{y}_{1:t-1})$, a single forward run of the SSPF algorithm proceeds as follows at the current time step.

ALGORITHM 2 SSPF Forward Run

• For $n = 1\ldots N$, initialise the cumulative weights $\alpha^{(n)}_{0,t} = w^{(n)}_{t-1}$.

• For $k = 1\ldots K$,

— For $n = 1\ldots N$, generate new samples for the target states $\mathbf{x}^{(n)}_{k,t}\sim q_k(\mathbf{x}_{k,t}\mid\mathbf{x}^{(n)}_{k,t-1},\mathbf{y}_t)$.

— For $i = 1\ldots N_o$, $n = 1\ldots N$, generate samples for the target to measurement associations $\tilde{r}^{i(n)}_{k,t}\sim q(\tilde{r}^i_{k,t}\mid\tilde{r}^{i(n)}_{1:k-1,t},\mathbf{x}^{(n)}_{k,t},\mathbf{y}^i_t)$.

— For $n = 1\ldots N$, update and normalise the cumulative weights

$$\alpha^{(n)}_{k,t}\propto\alpha^{(n)}_{k-1,t}\frac{p_k(\mathbf{x}^{(n)}_{k,t}\mid\mathbf{x}^{(n)}_{k,t-1})}{q_k(\mathbf{x}^{(n)}_{k,t}\mid\mathbf{x}^{(n)}_{k,t-1},\mathbf{y}_t)}\cdot\prod_{i=1}^{N_o}\frac{p^i(\mathbf{y}^i_{\tilde{r}^{i(n)}_{k,t},t}\mid\mathbf{x}^{(n)}_{k,t})p(\tilde{r}^{i(n)}_{k,t}\mid\tilde{r}^{i(n)}_{1:k-1,t})}{q(\tilde{r}^{i(n)}_{k,t}\mid\tilde{r}^{i(n)}_{1:k-1,t},\mathbf{x}^{(n)}_{k,t},\mathbf{y}^i_t)},\qquad\sum_{n=1}^{N}\alpha^{(n)}_{k,t}=1. \tag{57}$$

— If resampling is required then for $n = 1\ldots N$, sample an index $m(n)\sim\{\alpha^{(l)}_{k,t}\}^N_{l=1}$ and replace $\{\alpha^{(n)}_{k,t},\mathbf{x}^{(n)}_{1:k,t},\{\tilde{r}^{i(n)}_{1:k,t}\}^{N_o}_{i=1}\}\leftarrow\{N^{-1},\mathbf{x}^{m(n)}_{1:k,t},\{\tilde{r}^{im(n)}_{1:k,t}\}^{N_o}_{i=1}\}$.

• For $n = 1\ldots N$, compute and normalise the particle weights

$$w^{(n)}_t\propto\alpha^{(n)}_{K,t}\left[\prod_{i=1}^{N_o}(V^i)^{-M_C^{i(n)}}p(M_C^{i(n)})\right],\qquad\sum_{n=1}^{N}w^{(n)}_t=1. \tag{58}$$

• If resampling is required then for $n = 1\ldots N$, sample an index $m(n)\sim\{w^{(l)}_t\}^N_{l=1}$ and replace $\{w^{(n)}_t,\mathbf{x}^{(n)}_t,\tilde{\lambda}^{(n)}_t\}\leftarrow\{N^{-1},\mathbf{x}^{m(n)}_t,\tilde{\lambda}^{m(n)}_t\}$.

Note that in the above $M_C^{i(n)}$ is deterministically determined once the association vector $\tilde{\mathbf{r}}^{i(n)}_t$ has been sampled. The resulting sample set $\{w^{(n)}_t,\mathbf{x}^{(n)}_t,\tilde{\lambda}^{(n)}_t\}^N_{n=1}$ is approximately distributed according to the joint filtering distribution at the current time step $p(\mathbf{x}_t,\tilde{\lambda}_t\mid\mathbf{y}_{1:t})$. Since the forward run has to be repeated a number of times at each time step computational savings can be achieved by performing each forward run on a reduced particle set. This reduced set can be obtained by resampling with replacement from the input particle set.

2) *Association Proposal*: In this section we develop an efficient proposal for the target to measurement associations, and show how to sample from it. The proposal, which is similar in spirit to the one developed in [36], is based on the notion of a soft-gating of the measurements. The general form of the proposal has already been alluded to in the previous section where we discussed the SSPF architecture. Recall from (53) that it depends only on information available at the current time step, and that it factorises over the individual observers. For notational clarity we thus suppress the time and observer indices here, and focus only on the proposal for a single observer, with those for the other observers following in a similar fashion. Recall also from the previous section that the proposal is defined in terms of the target to measurement association vector, with the proposals for the number of clutter measurements and target detections being implicit. As discussed in the previous section the proposal for the association vector is assumed to take the following factorised form

$$q(\tilde{\mathbf{r}}\mid\mathbf{x},\mathbf{y})=\prod_{k=1}^{K}q(\tilde{r}_k\mid\tilde{r}_{1:k-1},\mathbf{x}_k,\mathbf{y}). \tag{59}$$

Recall from the discussion in the previous section that the factorisation can be performed over any permutation of the target ordering.

The components of the association vector are sampled sequentially conditional on each other. The proposal for the kth component is conditional on all the components sampled earlier in the sequence. We make use of this property to ensure that measurements associated with targets earlier in the sequence are not considered as candidates to be associated with the current target. In this way the algorithm is guaranteed to generate only valid association hypotheses. The probability of a particular target to measurement association should be high for those measurements close to the target, and should diminish as the distance between the measurement and target increases. For targets with no measurements in their immediate vicinity the probability of being undetected should be significant. These requirements can be achieved by using Bayes' rule to define the proposal for the kth component of the association vector as

$$q(\tilde{r}_k = j \mid \tilde{r}_{1:k-1}, \mathbf{x}_k, \mathbf{y}) = \frac{q(\mathbf{y} \mid \tilde{r}_k = j, \mathbf{x}_k) q(\tilde{r}_k = j \mid \tilde{r}_{1:k-1}, \mathbf{x}_k)}{\sum_{l=0}^{M} q(\mathbf{y} \mid \tilde{r}_k = l, \mathbf{x}_k) q(\tilde{r}_k = l \mid \tilde{r}_{1:k-1}, \mathbf{x}_k)}. \tag{60}$$

We assume the data component of the proposal to factorise over the individual measurements, with the unassigned measurements following a uniform clutter model, and the assigned measurement to be generated by the relevant likelihood model, i.e.,

$$q(\mathbf{y} \mid \tilde{r}_k = j, \mathbf{x}_k) = \begin{cases} V^{-M} & \text{if} \quad j = 0 \\ V^{-(M-1)} p_T(\mathbf{y}_j \mid \mathbf{x}_k) & \text{if} \quad j \in \{1 \ldots M\}. \end{cases} \tag{61}$$

We further set the prior component of the proposal to be equal to the factorised form of the target to measurement association prior in (23), i.e.,

$$q_j = q(\tilde{r}_k = j \mid \tilde{r}_{1:k-1}, \mathbf{x}_k) = p(\tilde{r}_k = j \mid \tilde{r}_{1:k-1}). \tag{62}$$

Thus each target has a non-zero probability of being undetected, and previously assigned measurements are prevented from being considered as candidates for association to the current target. With these definitions for the data and prior components the proposal for the kth component of the association vector can finally be expressed as

$$q(\tilde{r}_k = j \mid \tilde{r}_{1:k-1}, \mathbf{x}_k, \mathbf{y}) = \begin{cases} \dfrac{q_0 V^{-1}}{q_0 V^{-1} + \sum_{l=1}^{M} q_l p_T(\mathbf{y}_l \mid \mathbf{x}_k)} & \text{if} \quad j = 0 \\ \dfrac{q_j p_T(\mathbf{y}_j \mid \mathbf{x}_k)}{q_0 V^{-1} + \sum_{l=1}^{M} q_l p_T(\mathbf{y}_l \mid \mathbf{x}_k)} & \text{if} \quad j \in \{1 \ldots M\}. \end{cases} \tag{63}$$

Since this distribution is discrete it can easily be sampled from using standard techniques. Generating a sample for the entire association vector can be achieved by sequentially sampling the individual components conditional on each other from $\tilde{r}_1$ to $\tilde{r}_K$.

C. Independent Partition Particle Filter

Posterior dependencies between the targets exist due to the unknown association hypothesis. These dependencies can be removed by assuming the associations to be independent over the individual targets. This simplification is similar to the one made by the PMHT, and implies that any particular measurement can potentially be assigned to more than one target. As we illustrate below such an assumption facilitates an efficient component-wise sampling strategy to construct new joint particles. We refer to this algorithm as the IPPF. A similar strategy has been proposed in a somewhat different context before in [40].

More precisely, the PMHT simplification can be achieved by redefining the association prior in (23) as

$$p(\tilde{r}_k = j \mid \tilde{r}_{1:k-1}) = p(\tilde{r}_k = j) \propto \begin{cases} 1 - P_D & \text{if} \quad j = 0 \\ \dfrac{P_D}{K} & \text{otherwise} \end{cases} \tag{64}$$

where we have suppressed the time and observer indices for notational clarity. A prior of this form further implies that the proposals for the individual target associations in (63) also become independent, i.e., $q(\tilde{r}_k \mid \tilde{r}_{1:k-1}, \mathbf{x}_k, \mathbf{y}) = q(\tilde{r}_k \mid \mathbf{x}_k, \mathbf{y})$. Thus the product of the individual target weights in (56) now capture an independent factorisation over both the targets and the associations.

This factorisation immediately suggests an efficient component-wise sampling strategy to construct new particles. A new joint particle can be constructed by sampling the individual target components from the pool of particles, after the state prediction step, according to the individual target component weights. Thus the joint state-space is effectively partitioned so that the algorithm does not suffer from the curse of dimensionality. To ensure a properly weighted sample the final particle weight should be set to

$$w_t \propto w_{t-1} \left[\prod_{i=1}^{N_o} (V^i)^{-M_C^i} p(M_C^i) \right]. \tag{65}$$

A further resampling step on the joint particle can be included should these weights become degenerate.

The IPPF algorithm is summarised below. Assuming that the sample set $\{w_{t-1}^{(n)}, \mathbf{x}_{t-1}^{(n)}\}_{n-1}^{N}$ is approximately distributed according to the marginal filtering distribution at the previous time step $p(\mathbf{x}_{t-1} \mid \mathbf{y}_{1:t-1})$, the IPPF algorithm proceeds as follows at the current time step.

ALGORITHM 3 IPPF

• For $k = 1 \dots K$, $n = 1 \dots N$, generate new samples for the target states $\mathbf{x}_{k,t}^{(n)} \sim q_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}^{(n)}, \mathbf{y}_t)$.

• For $k = 1 \dots K$, $n = 1 \dots N$, $i = 1 \dots N_o$, generate samples for the target to measurement associations $\tilde{r}_{k,t}^{i(n)} \sim q(\tilde{r}_{k,t}^{i} \mid \mathbf{x}_{k,t}^{(n)}, \mathbf{y}_t^i)$.

• For $k = 1 \dots K$, $n = 1 \dots N$, compute and normalise the individual target weights

$$w_{k,t}^{(n)} \propto \frac{p_k(\mathbf{x}_{k,t}^{(n)} \mid \mathbf{x}_{k,t-1}^{(n)})}{q_k(\mathbf{x}_{k,t}^{(n)} \mid \mathbf{x}_{k,t-1}^{(n)}, \mathbf{y}_t)} \cdot \prod_{i=1}^{N_o} \frac{p^i(\mathbf{y}_{\tilde{r}_{k,t}^{i(n)},t}^{i} \mid \mathbf{x}_{k,t}^{(n)}) p(\tilde{r}_{k,t}^{i(n)})}{q(\tilde{r}_{k,t}^{i(n)} \mid \mathbf{x}_{k,t}^{(n)}, \mathbf{y}_t^i)} \tag{66}$$

$$\sum_{n=1}^{N} w_{k,t}^{(n)} = 1.$$

• For $k = 1 \dots K$, $n = 1 \dots N$, sample an index $m_k(n) \sim \{w_{k,t}^{(l)}\}_{l=1}^{N}$ and replace $\{\mathbf{x}_{k,t}^{(n)}, \{\tilde{r}_{k,t}^{i(n)}\}_{i=1}^{N_o}\} \leftarrow \{\mathbf{x}_{k,t}^{m_k(n)}, \{\tilde{r}_{k,t}^{im_k(n)}\}_{i=1}^{N_o}\}$.

• For $n = 1 \dots N$, compute and normalise the particle weights

$$w_t^{(n)} \propto w_{t-1}^{(n)} \left[\prod_{i=1}^{N_o} (V^i)^{-M_C^{i(n)}} p(M_C^{i(n)}) \right], \qquad \sum_{n=1}^{N} w_t^{(n)} = 1. \tag{67}$$

• If resampling is required then for $n = 1 \dots N$, sample an index $m(n) \sim \{w_t^{(l)}\}_{l=1}^{N}$ and replace $\{w_t^{(n)}, \mathbf{x}_t^{(n)}, \tilde{\lambda}_t^{(n)}\} \leftarrow \{N^{-1}, \mathbf{x}_t^{m(n)}, \tilde{\lambda}_t^{m(n)}\}$.

Note again that $M_C^{i(n)}$ is deterministically determined once the association vector $\tilde{\mathbf{r}}_t^{i(n)}$ has been sampled. The resulting sample set $\{w_t^{(n)}, \mathbf{x}_t^{(n)}, \tilde{\lambda}_t^{(n)}\}_{n=1}^{N}$ is approximately distributed according to the joint filtering distribution at the current time step $p(\mathbf{x}_t, \tilde{\lambda}_t \mid \mathbf{y}_{1:t})$, under the PMHT assumption.

VI. TARGET STATE PROPOSAL

Both the MC-JPDAF in Section IV and the multi-target particle filtering algorithms in Section V require the specification of a proposal distribution for the individual target states $q_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}, \mathbf{y}_t)$. This section presents and discusses some choices for this distribution.

In the original formulation of Monte Carlo techniques for sequential estimation in [42] and [43] the state proposal distribution was taken to be the target dynamics, i.e.,

$$q_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}, \mathbf{y}_t) = p_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}). \tag{68}$$

This is also the choice made in the first modern variant of the particle filter, known as the bootstrap filter [8]. Subsequently many particle filter practitioners have adopted this choice, since it leads to an intuitively simple strategy where new particles are predicted from the target dynamics, and the importance weights are proportional to the corresponding particle likelihoods. It can, however, lead to inefficient algorithms, since the state-space is explored without any knowledge of the observations. A standard approach to address this shortcoming is to increase the state noise for the proposal. This, however, leads to estimated trajectories that are less smooth than those predicted by the true target dynamics.

In [10] and [44] it is shown that the proposal distribution that is optimal in the sense that it minimises the variance of the importance weights is of the form

$$q_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}, \mathbf{y}_t) = p_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}, \mathbf{y}_t) \propto p(\mathbf{y}_t \mid \mathbf{x}_{k,t}) p_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}) \tag{69}$$

where $p(\mathbf{y}_t \mid \mathbf{x}_{k,t})$ is the likelihood conditional on the target state only. For models with nonlinearities, non-Gaussian noise, and data association uncertainty it is generally not possible to obtain a closed-form expression for the optimal proposal distribution.

As a compromise between the prior proposal and the optimal proposal we define a mixture proposal of the form

$$q_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}, \mathbf{y}_t) = \gamma^D p_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}) + (1 - \gamma^D) \sum_{(i,j) \in \mathcal{P}_k} \gamma_{i,j}^M q_k^i(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}, \mathbf{y}_{j,t}^i). \tag{70}$$

A proportion of γ^D, $0 \leq \gamma^D \leq 1$, of the new particles is sampled from the target dynamics. The remainder of the new particles is sampled from a mixture with each component accounting for one particular measurement. The set $\mathcal{P}_k$ contains the observer and measurement index pairs of those measurements deemed to have an impact on the state of the kth target. It can be found using a gating procedure similar to the one described in Section IVC, but using the particles approximating the filtering distribution at the previous time step. The mixture weights should sum to one, i.e., $\sum_{(i,j) \in \mathcal{P}_k} \gamma_{i,j}^M = 1$, and can be set to be inversely proportional to the distance between the transformed target state and the measurement under consideration. However, we normally set these weights to be uniform. We assume each individual mixture component to be of the form

$$q_k^i(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}, \mathbf{y}_{j,t}^i) \propto p_T^i(\mathbf{y}_{j,t}^i \mid \mathbf{x}_{k,t}) p_k(\mathbf{x}_{k,t} \mid \mathbf{x}_{k,t-1}) \tag{71}$$

where $p_T^i(\mathbf{y}_{j,t}^i \mid \mathbf{x}_{k,t})$ is the target likelihood at the ith observer. The component proposal is thus equal to the optimal proposal for the particular measurement, i.e., for the assignment $k = r_{j,t}^i$. However, even though each component concerns only a single measurement it is still not possible to obtain a closed-form result in general. In these cases we aim to find the best approximation to the optimal proposal within a given parametric class of distributions, such as the Gaussian

or Student-t distributions (see e.g. [10] for more detail on approximation techniques). This procedure is further exemplified in the example below.

EXAMPLE 3 This example follows on from Examples 1 and 2, and shows how to construct a Gaussian approximation to the optimal proposal in (71) by linearising the observation model (see also [10]). In what follows we will suppress the time t, target k, observer i, and measurement j indices for notational clarity, and denote the old state by $\mathbf{x}'$. Using the notation of Examples 1 and 2 the dynamic and likelihood models can be written as

$$\begin{aligned} p(\mathbf{x} \mid \mathbf{x}') &= \mathcal{N}(\mathbf{x} \mid \mathbf{A}\mathbf{x}', \boldsymbol{\Sigma}) \\ p_T(\mathbf{y} \mid \mathbf{x}) &= \mathcal{N}(\mathbf{y} \mid \mathbf{g}(\mathbf{x}, \mathbf{p}_o), \boldsymbol{\Sigma}_{\mathbf{y}}). \end{aligned} \tag{72}$$

Performing a first-order Taylor series expansion of the nonlinear mapping $\mathbf{g}$ around the point $\mathbf{x}^\star$ leads to a Gaussian approximation of the optimal proposal of the form

$$q(\mathbf{x} \mid \mathbf{x}', \mathbf{y}) = \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}^\star, \boldsymbol{\Sigma}^\star) \tag{73}$$

with

$$\begin{aligned} \boldsymbol{\Sigma}^\star &= (\boldsymbol{\Sigma}^{-1} + \mathbf{J}^{\star\mathrm{T}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{J}^\star)^{-1} \\ \boldsymbol{\mu}^\star &= \boldsymbol{\Sigma}^\star (\boldsymbol{\Sigma}^{-1} \mathbf{A}\mathbf{x}' + \mathbf{J}^{\star\mathrm{T}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} (\mathbf{y} - \mathbf{c}^\star)) \\ \mathbf{c}^\star &= \mathbf{g}(\mathbf{x}^\star, \mathbf{p}_o) - \mathbf{J}^\star \mathbf{x}^\star. \end{aligned} \tag{74}$$

In the above $\mathbf{J}^\star$ is the Jacobian of the nonlinear mapping $\mathbf{g}$ evaluated at the point $\mathbf{x}^\star$, and is given by

$$\mathbf{J}^\star = \hat{R}^{\star-1} \begin{bmatrix} (x^\star - x_o) & 0 & (y^\star - y_o) & 0 \\ -\hat{R}^{\star-1}(y^\star - y_o) & 0 & \hat{R}^{\star-1}(x^\star - x_o) & 0 \end{bmatrix} \tag{75}$$

where $\hat{R}^\star = ((x^\star - x_o)^2 + (y^\star - y_o)^2)^{1/2}$ is the target range. The point of expansion is normally taken to be the deterministic component of the state prediction, i.e., $\mathbf{x}^\star = \mathbf{A}\mathbf{x}'$.

VII. EXPERIMENTS AND RESULTS

In this section we evaluate and compare the performance of the proposed multi-target tracking algorithms on a challenging synthetic tracking problem. In what follows all location and distance measures are in metres, all angle measures in radians, all time measures in seconds, and all velocity measures in metres per second. We are interested in tracking slowly manoeuvring targets in the xy plane. We model each target with the near constant velocity model of Example 1, with $\sigma_x = \sigma_y = 5 \times 10^{-4}$ for all the targets. The discretisation time step for the model is set to $T = 1$. We track the targets from two sensors of the form described in Example 2, with $\sigma_R = 5$, $\sigma_\theta = 0.05$, and $R_{\max} = 150$ for both sensors. The sensors are located at $(-45, -45)$ and $(45, 45)$, respectively.

We consider tracking scenarios with $K = 3$ and $K = 4$ targets. For the three target scenario the initial target positions and velocities are given by $(-50, 50)$, $(-50, 0)$, $(-50, -50)$ and $(1.0, -1.5)$, $(1.0, 0.0)$, $(1.0, 0.75)$, respectively. For the four target scenario the first three targets are identical to the three target scenario, with the initial position and velocity of the fourth target given by $(0, 50)$ and $(0.0, -1.5)$, respectively. For each scenario we evaluate the tracking algorithms under three sets of conditions, with increasing difficulty: an easy setting ($P_D = 1.0$, $\lambda_C = 0.5$), a medium setting ($P_D = 0.8$, $\lambda_C = 2.0$), and a difficult setting ($P_D = 0.5$, $\lambda_C = 5.0$). The detection probability P_D and the clutter rate λ_C are assumed to be the same for both sensors. For each setting we generate $L = 100$ time steps of artificial data by simulating directly from the target and sensor models as specified above.

For each setting we run the algorithms with an increasing number of particles, i.e., $N = 10$, 50, 100, 200, 500, 1000, and repeat each experiment 20 times to get a statistical reflection of the behaviour of the algorithms. We benchmark the performance of the three multi-target tracking algorithms against the standard particle filter. Gating is applied in the MC-JPDAF. All the algorithms are initialised with Gaussians around the true initial target states. For all the algorithms the proposal distribution for the state is taken to be the approximately optimal proposal of Example 3, with $\sigma_x = \sigma_y = 5 \times 10^{-2}$ and the sensor parameters as before. The standard particle filter uses the same association proposal as that for the SSPF defined in Section VB2, with the association proposal for the IPPF defined in Section VC. The resampling procedure, where applicable, is invoked as soon as the effective sample size drops below half the actual sample size N. For the SSPF three forward runs were performed at each time step, with the sample size reduced by half prior to each run. Using the particles we computed MMSE estimates for the states and their covariances as

$$\hat{\mathbf{x}}_t = \sum_{n=1}^{N} w_t^{(n)} \mathbf{x}_t^{(n)}, \qquad \hat{\boldsymbol{\Sigma}}_t = \sum_{n=1}^{N} w_t^{(n)} (\mathbf{x}_t^{(n)} - \hat{\mathbf{x}}_t)(\mathbf{x}_t^{(n)} - \hat{\mathbf{x}}_t)^{\mathrm{T}}. \tag{76}$$

Note that for the MC-JPDAF the MMSE estimates are computed individually for each of the targets using the appropriate marginal weights.

The root mean squared error (RMSE) statistics are depicted in Fig. 3. For each setting and a particular number of particles the graphs show the mean and the standard deviation of the RMSE over the 20 repetitions of the experiment. The RMSE for a single experiment was computed as

$$\mathrm{RMSE} = \sqrt{\frac{1}{L} \sum_{t=1}^{L} \|\hat{\mathbf{x}}_t - \mathbf{x}_t^\star\|^2} \tag{77}$$

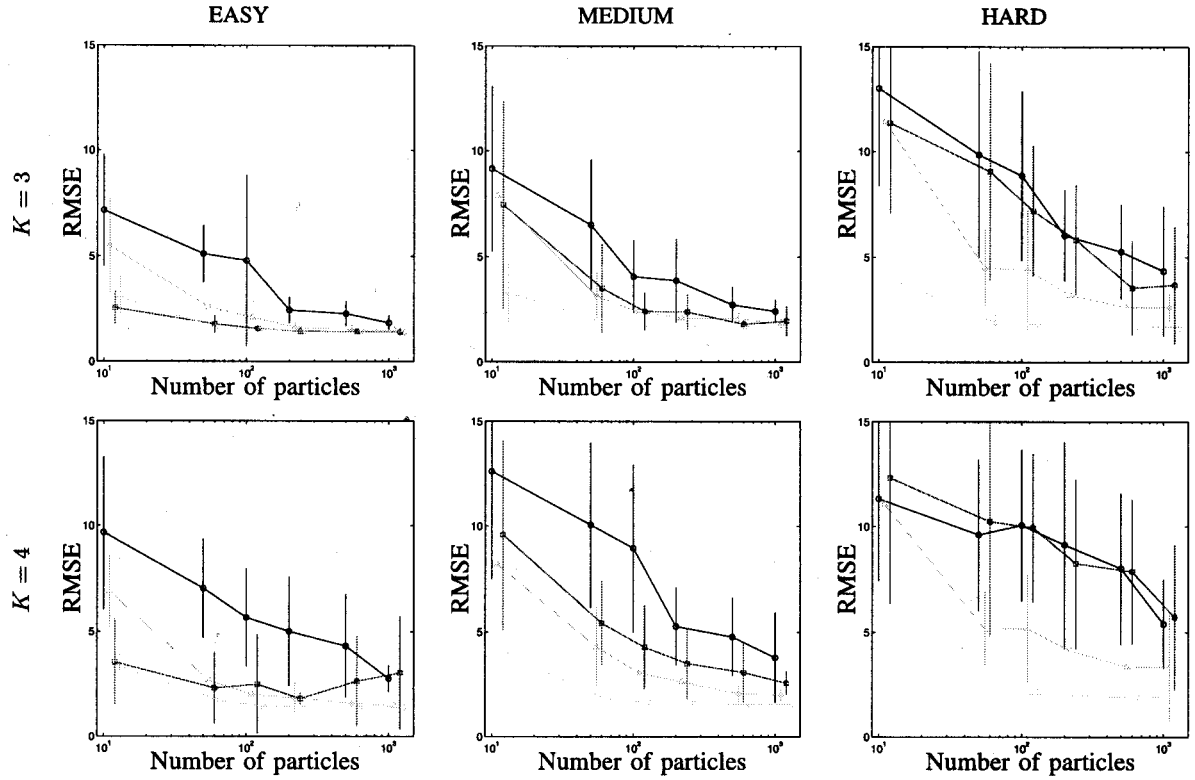


Fig. 3. RMSE error statistics. RMSE error (in metres) for standard particle filter (circles), SSPF (triangles), IPPF (squares), and MC-JPDAF (upside-down triangles). Left column is for easy problem ($P_D = 1.0$, $\lambda_C = 0.5$), middle column for medium problem ($P_D = 0.8$, $\lambda_C = 2.0$), right column for hard problem ($P_D = 0.5$, $\lambda_C = 5.0$). Top row is for $K = 3$ targets and bottom row is for $K = 4$ targets.

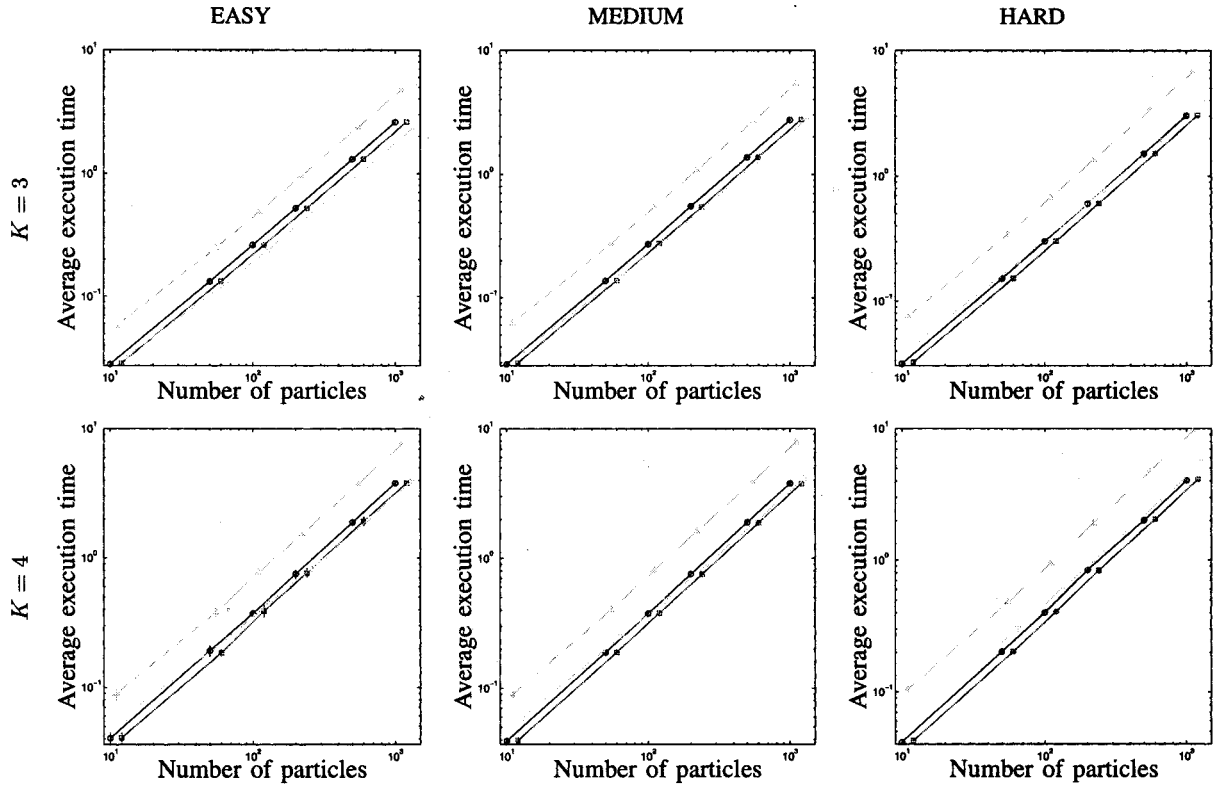


Fig. 4. Average execution time statistics. Average execution time (in seconds) per time step for standard particle filter (circles), SSPF (triangles), IPPF (squares), and MC-JPDAF (upside-down triangles). Left column is for easy problem ($P_D = 1.0$, $\lambda_C = 0.5$), middle column for medium problem ($P_D = 0.8$, $\lambda_C = 2.0$), right column for hard problem ($P_D = 0.5$, $\lambda_C = 5.0$). Top row is for $K = 3$ targets and bottom row is for $K = 4$ targets.

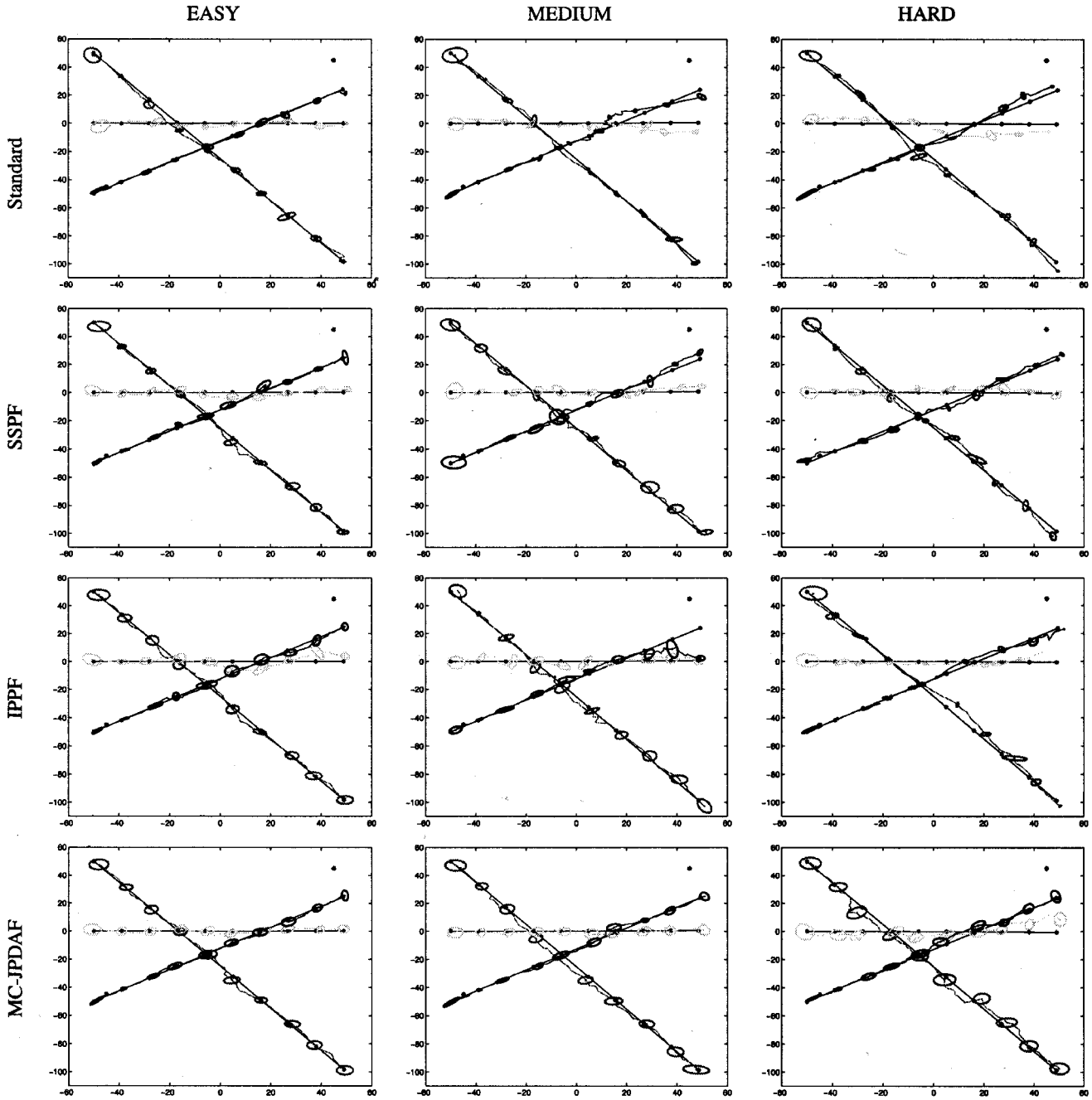


Fig. 5. Example trajectories for three targets. True (stars) and estimated (ellipses) trajectories for $K = 3$ targets and $N = 200$ particles. Ellipses indicate the 2-σ region of corresponding estimate covariances. Left column is for easy problem ($P_D = 1.0$, $\lambda_C = 0.5$), middle column for medium problem ($P_D = 0.8$, $\lambda_C = 2.0$), right column for hard problem ($P_D = 0.5$, $\lambda_C = 5.0$). Top row is for standard particle filter, second row for SSPF, third row for IPPF, and bottom row for MC-JPDAF. Dots indicate observer locations.

where $\mathbf{x}_t^\star$ is the true state at time t. As expected the error generally decreases with an increase in the number of particles, and appears to converge to a fixed value. An exception occurs for the IPPF for the experiments on the easy problem with $K = 4$ targets. Due to the PMHT assumption the IPPF was unable to disambiguate all the targets despite an increase in the number of particles. This is further exemplified by the example trajectories in Fig. 6 (third row, first column).

The standard particle filter, which searches directly in the joint space, is consistently outperformed by the other algorithms. This is due to the fact that the other algorithms all partition the space in some way or another to reduce the complexity of the search problem. The performance for the particle filtering strategies decreases as the problem becomes more difficult, in that more particles are required to achieve the same estimation accuracy. Note also that the relative performance of the IPPF rapidly degrades as

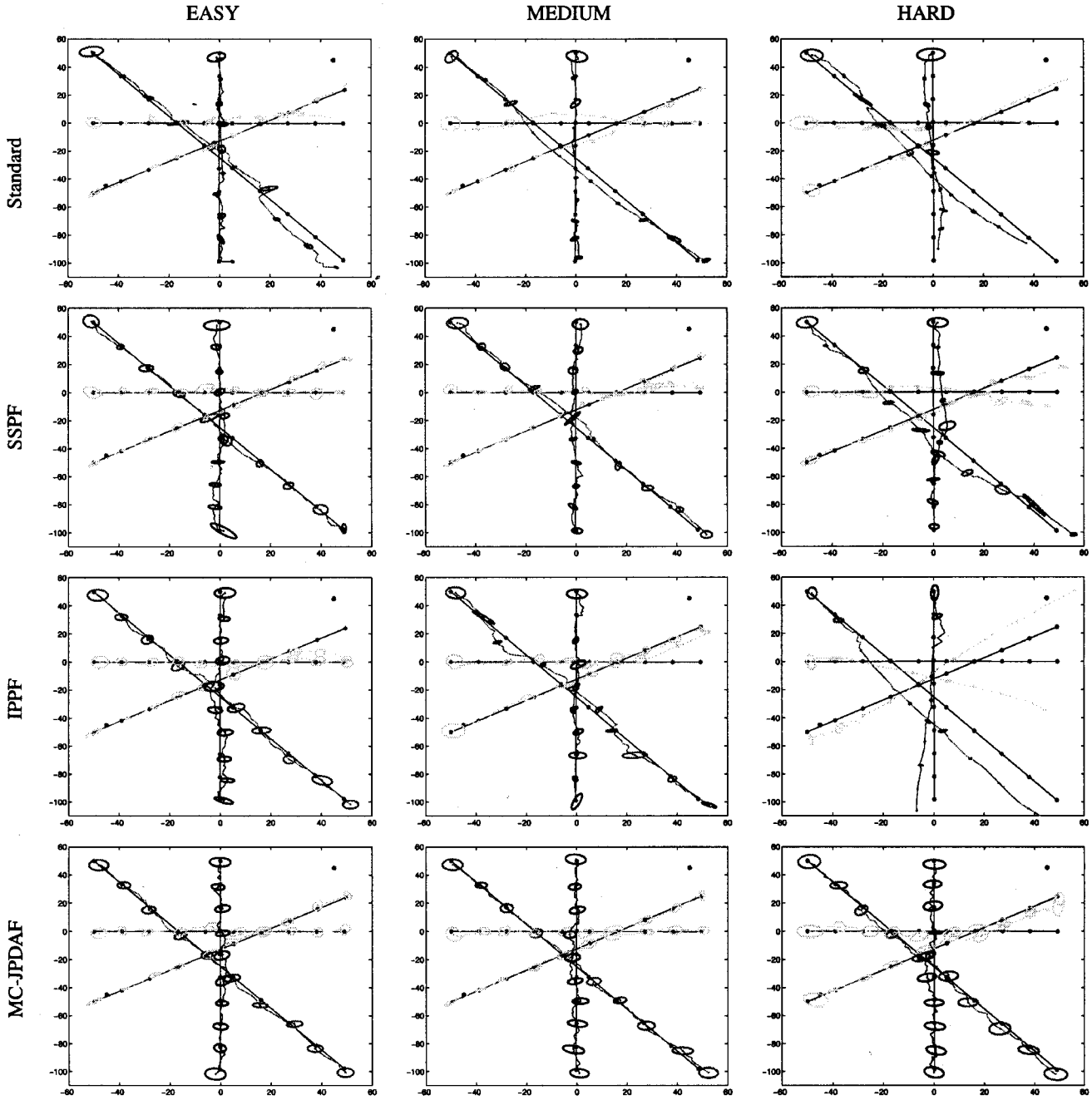


Fig. 6. Example trajectories for four targets. True (stars) and estimated (ellipses) trajectories for $K = 4$ targets and $N = 200$ particles. Ellipses indicate the 2-σ region of the corresponding estimate covariances. Left column is for easy problem ($P_D = 1.0$, $\lambda_C = 0.5$), middle column for medium problem ($P_D = 0.8$, $\lambda_C = 2.0$), right column for hard problem ($P_D = 0.5$, $\lambda_C = 5.0$). Top row is for standard particle filter, second row for SSPF, third row for IPPF, bottom row for MC-JPDAF. Dots indicate observer locations.

the problem difficulty increases. For example, for $K = 3$ targets it performs as well as the MC-JPDAF on the easy problem, similar to the SSPF on the medium problem, and just marginally better than the standard particle filter on the difficult problem. This provides further evidence that the PMHT assumption required to obtain the IPPF has an increasingly harmful effect as the association problem becomes more difficult.

The MC-JPDAF consistently outperforms the particle filtering strategies. What is even more remarkable is that it converges for a relatively small number of samples (between 100 and 200), and its performance does not appear to degrade significantly with an increase in the difficulty of the problem or the number of targets.

The average execution time statistics are depicted in Fig. 4. These are for a single time step of nonoptimised Matlab implementations for each of the algorithms. All the algorithms exhibit the same exponential trend. However, the slope is reasonably

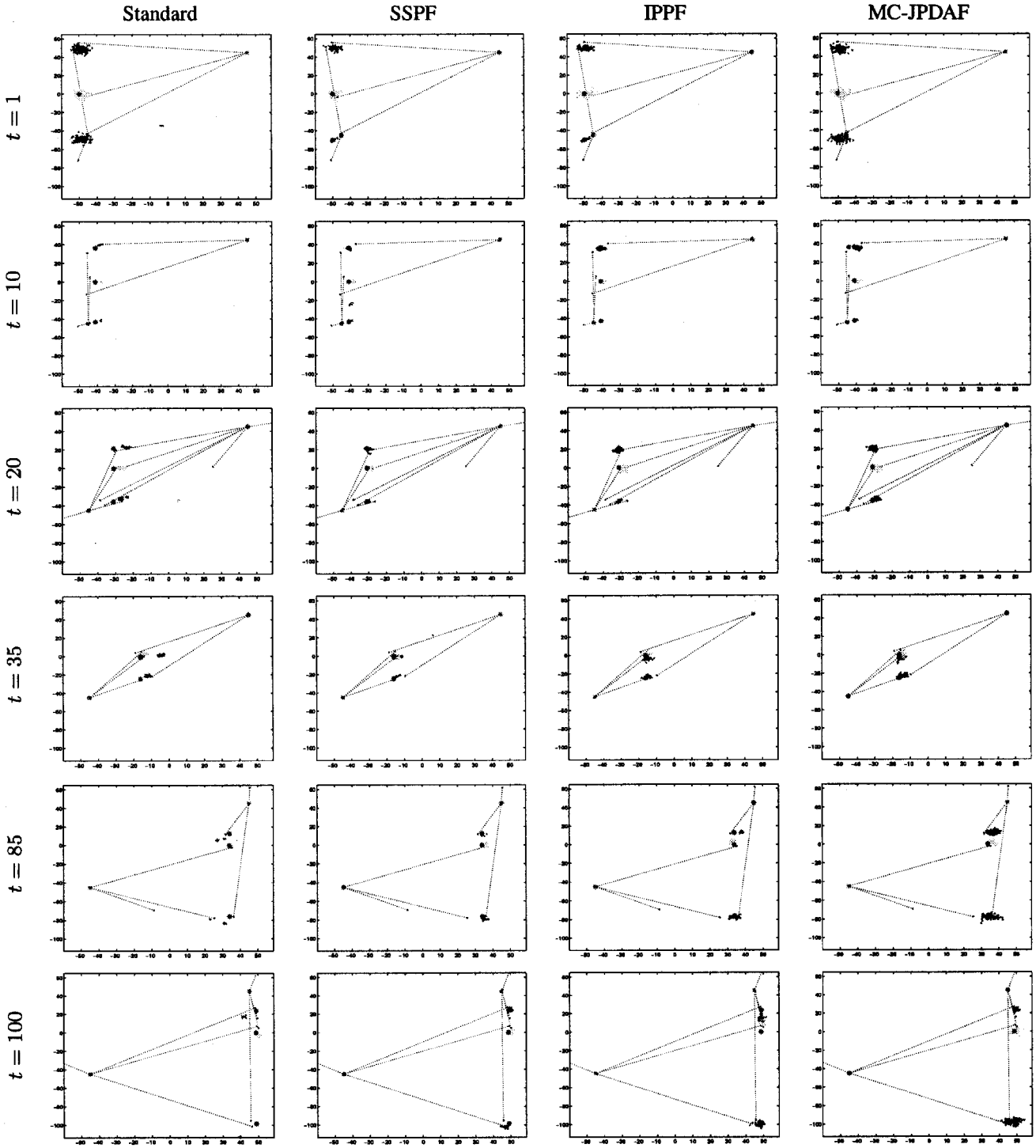


Fig. 7. Tracking snapshots. Snapshots of tracking for $K = 3$ targets and $N = 100$ particles. Detection probability and clutter rate were set to $P_D = 0.85$ and $\lambda_C = 1.0$, respectively. Left column is for standard particle filter, second column for SSPF, third column for IPPF, right column for MC-JPDAF. From top to bottom the rows correspond to snapshots taken at $t = 1, 10, 20, 35, 85, 100$ in a simulation of 100 time steps. The target locations are indicated by stars, and particle clouds for the individual targets by dots. Line of sight measurements from the stationary observers are indicated by dotted lines.

small, with the execution time increasing an order of magnitude for a two order of magnitude increase in the number of particles. Furthermore, most of the algorithms achieve an acceptable error performance while the average execution time per time step is well within the limits of practically realisable systems. The SSPF is computationally the most expensive, with the performance of the other algorithms being roughly similar. The computational complexity of the MC-JPDAF increases somewhat relative to the other

algorithms as the difficulty of the problem increases, due to the fact that more association hypotheses pass the gating test.

Some example trajectories for $K = 3$ and $K = 4$ targets are shown in Figs. 5 and 6, respectively. In all cases $N = 200$ particles were used. As expected the estimated trajectories become less accurate as the difficulty of the problem increases. Note that the estimated target locations are more accurate closer to the observers. This is especially evident for the blue target as it passes over the first observer. From the figure it appears as if the estimate covariances are smaller for the standard particle filter compared to those for the other algorithms. This does not imply that the standard particle filter yields more accurate estimates. Rather it is indicative of the sample depletion that occurs due to the global resampling procedure employed by the standard particle filter. Note also that both the IPPF and the MC-JPDAF are not always able to disambiguate all the targets.

The sample depletion problem is further exemplified by the tracking snapshots in Fig. 7. These results are for $K = 3$ targets and $N = 100$ particles, with the target detection probability and clutter rate set to $P_D = 0.85$ and $\lambda_C = 1.0$, respectively. The global resampling procedure leads to a largely depleted sample representation for the standard particle filter after as few as 10 time steps. Due to the resulting inaccuracy in the sample representation track is lost and never fully recovered. The other particle filtering strategies are able to maintain richer representations, allowing them to successfully track all the targets. The richest representation, however, is achieved by the MC-JPDAF, for which the sample clouds consistently reflect the reasonable uncertainty in the problem, without being adversely affected by the resampling procedure. Note also here the failure of the IPPF to disambiguate all the targets (last row, third column).

VIII. CONCLUSIONS

In this paper we developed a number of strategies for multi-target tracking and data association. The methods are applicable to general nonlinear and non-Gaussian models. The first method is the MC-JPDAF, for which the distributions of interest are the marginal filtering distributions for each of the targets. As opposed to the standard JPDAF these are approximated with particles rather than Gaussians. We also presented two extensions to the standard particle filtering methodology for tracking multiple targets. The SSPF samples the individual targets sequentially by utilising a factorisation of the importance weights. The IPPF, on the other hand, further makes the assumption that the associations are independent over the individual targets, leading to an efficient component-wise sampling strategy to construct new joint particles. The proposed algorithms effectively combat the curse of dimensionality by partitioning the state-space, leading to a sequence of easier estimation problems in each of the resulting subspaces.

The algorithms were evaluated and benchmarked against the standard particle filter on a challenging synthetic tracking problem. For all of the proposed algorithms the tracking accuracy was superior to that of the standard particle filter, with a comparable computational cost. The performance of the IPPF degraded rapidly with an increase in the difficulty of the association problem, signifying the harmful effect of the independence assumption under conditions of a low detection probability and a high clutter rate. The MC-JPDAF outperformed the particle filtering strategies under all conditions. It consistently converged for a relatively small number of particles, and its performance did not degrade significantly with an increase in the difficulty of the association problem or the number of targets.

A number of open questions remain, and various avenues are available for future research. Two very pertinent problems are the modification of the algorithms to deal with an unknown and variable number of targets, and the development of automatic initialisation (or detection) procedures. It may also be possible to improve the performance of the SSPF by incorporating the temporal smoothing ideas of [45] and [46]: after sampling the targets sequentially in the forward direction, a backward smoothing run may be designed to further refine the sample representations for the individual targets. The algorithms presented here can also be applied, with mild modifications, to the tracking of multi-part or extended objects [36, 47], or objects whose motion exhibit some degree of mutual correlation.

REFERENCES

[1] Schulz, D., Burgard, W., Fox, D., and Cremers, A. B.
Tracking multiple moving targets with a mobile robot using particle filters and statistical data association.
In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2001, 1665–1670.

[2] Schulz, D., Burgard, W., and Fox, D.
People tracking with mobile robots using sample-based joint probabilistic data association filters.
International Journal of Robotics Research, **22**, 2 (2003).

[3] Anderson, B. D. O., and Moore, J. B.
Optimal Filtering.
Englewood Cliffs, NJ: Prentice-Hall, 1979.

[4] Julier, S. J., and Uhlmann, J. K.
A new extension of the Kalman filter to nonlinear systems.
In *Proceedings of AeroSense: The 11th International Symposium on Aerospace/Defence Sensing, Simulation and Controls, Vol. Multi Sensor Fusion, Tracking and Resource Management II*, 1997.

[5] Julier, S. J., and Uhlmann, J. K.
Unscented filtering and nonlinear estimation.
Proceedings of the IEEE, **92**, 3 (2004), 401–422.

[6] Alspach, D. L., and Sorenson, H. W.
Nonlinear Bayesian estimation using Gaussian sum approximation.
IEEE Transactions on Automatic Control, **17**, 4 (1972), 439–448.

[7] Kitagawa, G.
Non-Gaussian state-space modeling of nonstationary time series (with discussion).
Journal of the American Statistical Association, **82**, 400 (1987), 1032–1063.

[8] Gordon, N. J., Salmond, D. J., and Smith, A. F. M.
Novel approach to nonlinear/non-Gaussian Bayesian state estimation.
IEE Proceedings, Pt. F, **140**, 2 (1993), 107–113.

[9] Kitagawa, G.
Monte Carlo filter and smoother for non-Gaussian nonlinear state space models.
Journal of Computational and Graphical Statistics, **5**, 1 (1996), 1–25.

[10] Doucet, A., Godsill, S. J., and Andrieu, C.
On sequential Monte Carlo sampling methods for Bayesian filtering.
Statistics and Computing, **10**, 3 (2000), 197–208.

[11] Doucet, A., de Freitas, J. F. G., and Gordon, N. J. (Eds.)
Sequential Monte Carlo Methods in Practice.
New York: Springer-Verlag, 2001.

[12] Liu, J. S., and Chen, R.
Sequential Monte Carlo methods for dynamic systems.
Journal of the American Statistical Association, **93** (1998), 1032–1044.

[13] Isard, M., and Blake, A.
CONDENSATION—conditional density propagation for visual tracking.
International Journal of Computer Vision, **29**, 1 (1998), 5–28.

[14] Popp, R. L., Kirubarajan, T., and Pattipati, K. R.
Survey of assignment techniques for multitarget tracking.
In Y. Bar-Shalom and W. D. Blair (Eds.), *Multitarget/Multisensor Tracking: Applications and Advances III*, Dedham, MA: Artech House, 2000, ch. 2.

[15] Read, D.
An algorithm for tracking multiple targets.
IEEE Transactions on Automation and Control, **24**, 6 (1979), 84–90.

[16] Bar-Shalom, Y., and Fortmann, T. E.
Tracking and Data Association.
New York: Academic Press, 1988.

[17] Fortmann, T. E., Bar-Shalom, Y., and Scheffe, M.
Sonar tracking of multiple targets using joint probabilistic data association.
IEEE Journal of Oceanic Engineering, **8** (1983), 173–184.

[18] Pao, L. Y.
Multisensor multitarget mixture reduction algorithms for target tracking.
AIAA Journal of Guidance, Control and Dynamics, **17** (1994), 1205–1211.

[19] Salmond, D. J.
Mixture reduction algorithms for target tracking in clutter.
In O. E. Drummond (Ed.), *Signal and Data Processing of Small Targets, SPIE 1305*, (1990), 434–445.

[20] Gauvrit, H., Le Cadre, J-P., and Jauffret, C.
A formulation of multitarget tracking as an incomplete data problem.
IEEE Transactions on Aerospace and Electronic Systems, **33**, 4 (1997), 1242–1257.

[21] Streit, R. L., and Luginbuhl, T. E.
Maximum likelihood method for probabilistic multi-hypothesis tracking.
In O. E. Drummond (Ed.), *Signal and Data Processing of Small Targets, SPIE 2235*, (1994).

[22] Dempster, A. P., Laird, N. M., and Rubin, D. B.
Maximum likelihood from incomplete data via the EM algorithm.
Journal of the Royal Statistical Society, Series B, **39**, 1 (1977), 1–38.

[23] Willett, P., Ruan, Y., and Streit, R.
PMHT: Problems and some solutions.
IEEE Transactions on Aerospace and Electronic Systems, **38**, 3 (2002), 738–754.

[24] Frank, O., Nieto, J., Guivant, J., and Scheding, S.
Multiple target tracking using sequential Monte Carlo methods and statistical data association.
In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2003.

[25] Karlsson, R., and Gustafsson, F.
Monte Carlo data association for multiple target tracking.
In *Proceedings of the IEE Seminar—Target Tracking: Algorithms and Applications*, 2001, 13/1–13/5.

[26] Avitzour, D.
Stochastic simulation Bayesian approach to multitarget tracking.
IEE Proceedings on Radar and Sonar Navigation, **142**, 2 (1995), 41–44.

[27] Gordon, N. J.
A hybrid bootstrap filter for target tracking in clutter.
IEEE Transactions on Aerospace and Electronic Systems, **33**, 1 (1997), 353–358.

[28] Hue, C., Le Cadre, J-P., and Pérez, P.
Tracking multiple objects with particle filtering.
IEEE Transactions on Aerospace and Electronic Systems, **38**, 3 (2002), 791–812.

[29] Geman, S., and Geman, D.
Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images.
IEEE Transactions on Pattern Analysis and Machine Intelligence, **PAMI-6**, 6 (1984), 721–741.

[30] Dellaert, F., Seitz, S. M., Thorpe, C., and Thrun, S.
EM, MCMC, and chain flipping for structure from motion with unknown correspondence.
Machine Learning, **50**, 1–2 (2003), 45–71.

[31] Gilks, W. R., Richardson, S., and Spiegelhalter, D. J.
Markov Chain Monte Carlo in Practice.
London: Chapman and Hall, 1996.

[32] Gordon, N. J., and Doucet, A.
Sequential Monte Carlo for manoeuvering target tracking in clutter.
In O. E. Drummond (Ed.), *Signal and Data Processing of Small Targets, SPIE 3809*, (1999), 493–500.

[33] Doucet, A., Gordon, N. J., and Krishnamurthy, V.
Particle filters for state estimation of jump Markov linear systems.
IEEE Transactions on Signal Processing, **49**, 3 (2001), 613–624.

[34] Pitt, M. K., and Shephard, N.
Filtering via simulation: Auxiliary particle filter.
Journal of the American Statistical Association, **94** (1999), 590–599.

[35] Doucet, A., Vo, B., Andrieu, C., and Davy, M.
Particle filtering for multi-target tracking and sensor management.
In *Proceedings of the 5th International Conference on Information Fusion*, 2002.

[36] Ikoma, N., and Godsill, S. J.
Extended object tracking with unknown association, missing observations, and clutter using particle filters.
In *Proceedings of the 12th IEEE Workshop on Statistical Signal Processing*, 2003, 485–488.

[37] Boers, Y., Driessen, J. N., Verschure, F., Heemels, W. P. M. H., and Juloski, A.
A multi target track before detect application.
In *Proceedings of the IEEE Workshop on Multi-Object Tracking*, 2003.

[38] Kreucher, C., Kastella, K., and Hero, A. O., III
Multitarget tracking using a particle filter representation of the joint multitarget density.
Submitted to *IEEE Transactions on Aerospace and Electronic Systems*, 2004.

[39] Salmond, D. J., and Birch, H.
A particle filter for track-before-detect.
In *Proceedings of the American Control Conference*, 2001, 3755–3760.

[40] Orton, M., and Fitzgerald, W.
A Bayesian approach to tracking multiple targets using sensor arrays and particle filters.
IEEE Transactions on Signal Processing, **50**, 2 (2002), 216–223.

[41] Gordon, N. J., Salmond, D. J., and Fisher, D.
Bayesian target tracking after group pattern distortion.
In O. E. Drummond (Ed.), *Signal and Data Processing of Small Targets, SPIE 3163*, (1997), 238–248.

[42] Handschin, J. E., and Mayne, D. Q.
Monte Carlo techniques to estimate the conditional expectation in multi-stage non-linear filtering.
International Journal of Control, **9**, 5 (1969), 547–559.

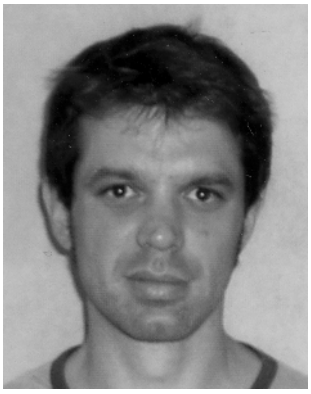
[43] Handschin, J. E.
Monte Carlo techniques for prediction and filtering of non-linear stochastic processes.
Automatica, **6** (1970), 555–563.

[44] Kong, A., Liu, J. S., and Wong, W. H.
Sequential imputations and Bayesian missing data problems.
Journal of the American Statistical Association, **89** (1994), 278–288.

[45] Fong, W., Godsill, S. J., Doucet, A., and West, M.
Monte Carlo smoothing with application to speech enhancement.
IEEE Transactions on Signal Processing, **50**, 2 (Feb. 2002), 438–449.

[46] Godsill, S. J., Doucet, A., and West, M.
Monte Carlo smoothing for non-linear time series.
Journal of the American Statistical Association, **99**, 465 (Mar. 2004), 156–168.

[47] Vermaak, J., Ikoma, N., and Godsill, S. J.
Extended object tracking using particle techniques.
In *Proceedings of the IEEE Aerospace Conference*, 2004.

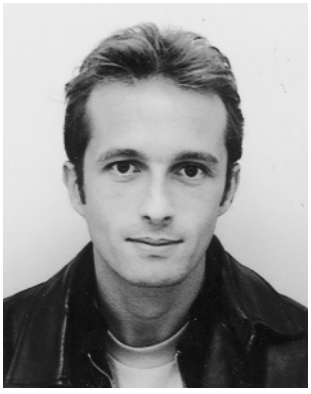


Jaco Vermaak was born in South Africa in 1969. He received the B.Eng. and M.Eng. degrees from the University of Pretoria, South Africa in 1993 and 1996, respectively, and the Ph.D. degree at the University of Cambridge, U.K. in 2000.

He worked as a post-doctoral researcher at Microsoft Research Europe, Cambridge, U.K. between 2000 and 2002. He is currently employed as a senior research associate in the Signal Processing Group of the Cambridge University Engineering Department. His research interests include audio-visual tracking techniques, multi-media manipulation, statistical signal processing methods and machine learning.

Simon J. Godsill (S'91—M'93) is reader in Statistical Signal Processing in the Engineering Department of Cambridge University. He is also a fellow of Corpus Christi College Cambridge. He has research interests in Bayesian and statistical methods for signal processing, Monte Carlo algorithms for Bayesian problems, modelling and enhancement of audio and musical signals, tracking, and genomic signal processing.

Dr. Godsill is an associate editor for the *IEEE Transactions on Signal Processing* and the *Journal Bayesian Analysis*, and is a member of the IEEE Signal Processing Theory and Methods Committee. He has published extensively in journals, books and conferences, acted as an editor for the IEEE and EURASIP for special journal issues on Monte Carlo methods, and regularly organises themed sessions at international conferences.



Patrick Pérez was born in 1968. He graduated from École Centrale Paris, France, in 1990 and received the Ph.D. degree from the University of Rennes, France, in 1993. After one year as an Inria post-doctoral researcher in the Department of Applied Mathematics at Brown University, Providence, RI, he was appointed at Inria in 1994 as a full time researcher. From 2000 to 2004, he was with Microsoft Research in Cambridge, U.K. In 2004, he became senior researcher at Inria, and he is now with the Vista research group at Irisa/Inria-Rennes. His research interests include probabilistic models for understanding, analysing, and manipulating still and moving images.