A Bayesian Approach to Multiple Target Detection and Tracking

Mark R. Morelande, Christopher M. Kreucher, and Keith Kastella

Abstract—This paper considers the problem of simultaneously detecting and tracking multiple targets. The problem can be formulated in a Bayesian framework and solved, in principle, by computation of the joint multitarget probability density (JMPD). In practice, exact computation of the JMPD is impossible, and the predominant challenge is to arrive at a computationally tractable approximation. A particle filtering scheme is developed for this purpose in which each particle is a hypothesis on the number of targets present and the states of those targets. The importance density for the particle filter is designed in such a way that the measurements can guide sampling of both the target number and the target states. Simulation results, with measurements generated from real target trajectories, demonstrate the ability of the proposed procedure to simultaneously detect and track ten targets with a reasonable sample size.

Index Terms—Bayes' procedures, object detection and tracking, particle filters.

I. INTRODUCTION

THE classical approach to multiple target tracking (MTT) is exemplified by the technique commonly referred to as multiple hypothesis tracking (MHT) [1], [2]. This technique typically works with a set of detections comprised of both noisy measurements of the target position, in Cartesian or polar coordinates, and false alarms due to clutter. The detections are then either associated with existing tracks, used to create new tracks, or deemed false alarms. Typically, Kalman-filter-type algorithms are used to update the existing tracks after association. Association between measurements and targets is done by enumerating and evaluating lists of feasible measurement-to-target associations, which are managed through the use of hypothesis management techniques.

A problem with this traditional approach to MTT is that information is removed when a set of detections is produced from a collection of raw sensor returns. This information reduction makes it difficult for the tracking algorithm to recognize and accurately track targets that rarely produce returns above the detection threshold. A more robust solution to MTT is possible if the raw sensor returns are made available to the tracker. This approach is commonly referred to as track-before-detect (TBD) [3]. Here, the surveillance region is divided into cells, and the tracking algorithm is supplied with the intensity of the sensor return in each cell. Early approaches to TBD include dynamic programming algorithms [4] and maximum-likelihood techniques [5]. These approaches are not recursive and have difficulty modelling complicated target motion. A more promising approach is to adopt a fully Bayesian perspective where the quantity of interest is a Markov process, the multitarget state, which is the concatenation of several individual target states. Stone [6] develops a mathematical theory of MTT from a Bayesian point of view. Srivistava, Miller [7], Kastella [8], and Mahler [9] also did early work in this area. The Bayesian approach has the advantages of providing a recursive solution with arbitrary target dynamic models.

This paper presents a unified approach to multitarget detection and tracking based on recursive approximation of the joint multitarget probability density (JMPD). The approach is developed for a pixelized measurement model similar to that used in TBD. The JMPD is a single probabilistic entity that captures uncertainty about the number of targets present in the surveillance region as well as their individual states. While developed independently [10], [8], the JMPD method can be derived using the mathematics of random sets and expressed in the finite-set statistics framework [9]. As discussed in [11], JMPD can be traced back to the event-averaged maximum-likelihood estimation (EAMLE) work of Kastella [8], [10] and many earlier works, e.g., [12]–[14].

Since the JMPD is a high-dimensional entity that cannot be computed in closed form, sophisticated numerical techniques are required to obtain a tractable approximation. Early work used a deterministic grid approximation, which is practical only for simple problems involving a small number of targets moving in one dimension [15]. More recently, particle filters (PFs) have been used to approximate the JMPD in realistic scenarios involving large numbers of targets moving in two-dimensions [16], [17]. PFs provide a recursive stochastic grid approximation to the exact solution of Bayesian state estimation problems [18], [19]. The idea is that, by simulating possible target trajectories and weighting them according to their suitability, a set of particles will be obtained that move of their own volition to the desired parts of the state space. This avoids the computational and implementational complexities of deterministic grid approximations [20]. PFs are often implemented using sequential importance sampling in which samples from the posterior are obtained by drawing from an importance density which factorizes in such a way that a sequential sampling procedure is obtained.

Efficient particle proposals, in the sense that many particles are placed in desirable parts of the state space, are necessary for tractable approximation using PFs. This is particularly true for MTT where the dimension of the state vector is proportional to the number of targets in the surveillance region. Previous PFs for MTT with pixelized measurement models have not addressed this issue. In [21], particle proposals are made using the state transition density, a notoriously inefficient scheme for any state dimension which quickly deteriorates as the state dimension increases. The lack of suitability of this scheme for MTT has been demonstrated in [17]. The algorithm of [17] uses measurement-directed proposals but samples inefficiently when multiple targets are in close proximity. The work presented here expands on [17] in two ways. First, an optimal importance sampling procedure, in the sense described in [22], which enables efficient tracking of persistent targets is developed. Second, the algorithm is extended to provide simultaneous multitarget detection and tracking.

The problem of tracking a known and fixed number of targets is considered first. This is an extension of the work presented in [23]. As in [17], the idea of clustering the multitarget state into groups of closely spaced targets is used. This procedure was originally suggested in [24] for MTT using sensor arrays where it was referred to as the independent partition PF. Clustering enables computationally tractable tracking of large numbers of targets. Unlike [17], in the technique proposed here, samples for each cluster of targets are drawn jointly enabling the filter to better handle situations in which several targets are in close proximity. This will be seen to have a dramatic effect on tracking performance. An important issue in this approach is the distance threshold used in the clustering. Selection of this threshold involves balancing conflicting requirements. This issue is investigated through the variances of the importance sampling weights. Rao–Blackwellization [25] is used to reduce the dimension of the PF.

If the number of targets is unknown and time varying, MTT is performed by computing, or numerically approximating, the JMPD of the extended multitarget state, a quantity that is the concatenation of the number of targets and the state of each target. Each particle in the PF approximation to the JMPD is therefore a hypothesis of the number of targets present and the state of each target. Thus, each particle can be comprised of a different number of individual target states. A model for the time evolution of the extended multitarget state is developed that allows the rates at which targets arrive and depart to vary both spatially and temporally. Sampling directly from this transition model does not lead to a tractable algorithm since, depending on the amount of prior information available, it is unlikely that a sufficient mass of particles will be placed in the correct part of the extended multitarget state space. Sampling is therefore performed from an importance density, which uses a measurement-directed method to decide which areas of the surveillance region are more likely to have had a target arrive or leave.

The rest of this paper is organized in the following manner. The target dynamic and measurement models along with some notation are outlined in Section II. In Section III, the JMPD is briefly introduced and it is shown how the rules of Bayesian filtering are applied to produce a recursive filtering procedure. A particle filter for track maintenance of persistent targets is given in Section IV. This filter is extended in Section V to include initiation and deletion of tracks in addition to maintenance of existing tracks. Simulation results examining the filter performance under various conditions are provided in both sections. Finally, Section VI contains a brief summary and discussion.

II. NOTATION AND MODELING

Consider the presence of r^t targets at time t with the state of the ith target denoted as $\mathbf{x}_i^t \in \mathbb{R}^{d_x}, i = 1, \ldots, r^t$. The multitarget state at time t is defined as the concatenation of the individual target states $\mathbf{X}^t = (\mathbf{x}_1^{t\prime}, \ldots, \mathbf{x}_{r^t}^{t\prime})'$, where $'$ denotes the matrix transpose. In general, the target number is unknown and time varying and can therefore be considered a state variable to be estimated with the individual target states. The extended multitarget state is defined as $\boldsymbol{\xi}^t = (\mathbf{X}^{t\prime}, r^t)'$. It is assumed the extended multitarget state is a Markov sequence satisfying

$$\boldsymbol{\xi}^t \,|\, \boldsymbol{\xi}^{t-1} \sim a^t, \quad t = 1, 2, \ldots, \quad (1)$$

where $z \sim p$ means that z is a random variable with probability density function p. It is important to keep in mind that (1) involves transitions in the target number in addition to the usual transitions of the individual target states.

In this paper, it will be assumed that each target moves in s-dimensional space with individual target states composed of position and velocity in each direction. The position elements of the ith target are collected into $\boldsymbol{\rho}_i^t$ and the velocity elements are collected into $\mathbf{v}_i^t$ so that $\mathbf{x}_i^t = (\boldsymbol{\rho}_i^{t\prime}, \mathbf{v}_i^{t\prime})'$. The process noise is assumed to be Gaussian distributed. Individual target states then evolve independently according to

$$\mathbf{x}_i^t \,|\, \mathbf{x}_i^{t-1} \sim N\left(\cdot\,; \mathbf{F}\mathbf{x}_i^{t-1}, \mathbf{Q}_i^t\right) \quad (2)$$

where $N(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the Gaussian probability density function evaluated at $\mathbf{x}$ with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}, \mathbf{Q}_i^t$ is the covariance matrix for the process noise of the ith target at time t and

$$\mathbf{F} = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix} \otimes \mathbf{I}_s \quad (3)$$

with $\mathbf{I}_m$ denoting the $m \times m$ identity matrix, $\otimes$ denoting the Kronecker product, and T denoting the sampling period. Modeling of target number transitions will be discussed in Section V. Equation (2) models the motion of a target moving with a velocity subject to random perturbations. If the intensity of the process noise is sufficiently large these perturbations can model sudden changes in target velocity, i.e., maneuvers. A better approach to tracking maneuverring targets is to allow the target to move in one of several possible motion modes. A Markov chain, referred to as the maneuvering mode, switches between these motion modes. The PF described in Sections IV and V can be extended to handle multiple motion models by adding the maneouvring mode to the target state and producing samples of this discrete random variable along with the other elements of the target state [26], [27].

Target states are observed using a procedure that distorts the target states in a probabilistically known manner

$$\mathbf{z}^t \mid \mathbf{X}^t \sim b^t, \quad t = 1, 2, \ldots, \tag{4}$$

where $\mathbf{z}^t$ is the vector of measurements at time t and the density b^t is a probabilistic description of the measurement procedure at time t. A description of the specific measurement model to be used in this paper will now be given.

The observation region $\Omega \subset \mathbb{R}^s$ is divided into J subregions, referred to as cells, with the vector $\mathbf{z}^t = (z_1^t, \ldots, z_J^t)'$ containing the measurements obtained in each subregion. The position elements of the target states are collected into $\mathbf{P}^t = (\boldsymbol{\rho}_1^{t\prime}, \ldots, \boldsymbol{\rho}_{r^t}^{t\prime})'$. Measurements are made independently in each cell with the distribution of the measurement in a cell depending on the number of targets in the cell. Note that a target affects only the measurement of the cell which it occupies. This makes the measurement model applicable primarily to low resolution sensors such as ground moving target indicator (GMTI) sensors in wide area search mode [28]. Although changes to the details of the algorithm would be necessary, it is anticipated that the general procedure proposed here would also be applicable to a high resolution sensor in which targets affect the values of multiple cell measurements.

Let $o_j(\mathbf{P}^t)$ denote the number of targets occupying the jth cell at time t. Then

$$b^t(\mathbf{z}^t \mid \mathbf{P}^t) = \prod_{j=1}^J l_{o_j(\mathbf{P}^t)}\left(z_j^t\right). \tag{5}$$

The development of the tracking algorithm does not require a particular form for the cell measurement likelihood l_m, although proper functioning of the algorithm will require that differences in the number of targets in a cell are reflected in the sensor return, e.g., $l_m(z)$ should place more probability mass in regions of large z as m increases. In this paper, Rayleigh-distributed measurements are assumed. Similar models have been used in monopulse radar applications [29] and to model clutter and target returns in turbulent environments [30]. In the case of nonthresholded measurements, $z_j^t \in [0, \infty)$ and the cell measurement likelihood is, for $m = 0, 1, 2, \ldots, z > 0$,

$$l_m(z) = \frac{z}{1 + \lambda m} \exp\left(-\frac{z}{2(1 + \lambda m)}\right) \tag{6}$$

where λ is referred to as the signal-to-noise ratio (SNR). For thresholded measurements, $z_j^t \in \{0, 1\}$ with $z_j^t = 1$ corresponding to a target detection in the jth cell at time t. The threshold is set so that $\Pr(z_j^t = 1 \mid o_j(\mathbf{P}^t) = 0) = P_{\text{FA}}$ where P_{FA} is the desired false alarm probability. Then, for $m = 0, 1, 2, \ldots,$

$$l_m(z) = \begin{cases} P_{\text{FA}}^{1/(1+m\lambda)}, & z = 1, \\ 1 - P_{\text{FA}}^{1/(1+m\lambda)}, & z = 0. \end{cases} \tag{7}$$

III. JOINT MULTITARGET PROBABILITY DENSITY FUNCTION

MTT involves recursive computation of the JMPD $p(\boldsymbol{\xi}^t \mid \mathbf{Z}^t)$, where $\boldsymbol{\xi}^t = (\mathbf{X}^{t\prime}, r^t)'$ is the extended multitarget state, composed of the target number and the collection of individual target

states, and $\mathbf{Z}^t = \{\mathbf{z}^1, \ldots, \mathbf{z}^t\}$ is the measurement history [15]. The JMPD provides a probabilistic description of the number of targets present and their states from which quantities of interest can be estimated. For instance, the posterior probability that r targets are present in region $\Theta \subset \mathbb{R}^{d_x}$ of the state space is $\int_{\Theta^r} p(\mathbf{X}^t, r^t = r \mid \mathbf{Z}^t) d\mathbf{X}^t$. Assuming $p(\boldsymbol{\xi}^{t-1} \mid \mathbf{Z}^{t-1})$ is available the JMPD at time t is updated via the recursion [13], [31]

$$\begin{aligned} p(\mathbf{X}^t, &r^t \mid \mathbf{Z}^t) \\ &= \frac{p(\mathbf{z}^t \mid \mathbf{X}^t, r^t)}{p(\mathbf{z}^t \mid \mathbf{Z}^{t-1})} \sum_{r^{t-1}=0}^{\infty} \int p(\mathbf{X}^t, r^t \mid \mathbf{X}^{t-1}, r^{t-1}) \\ &\quad \times p(\mathbf{X}^{t-1}, r^{t-1} \mid \mathbf{Z}^{t-1}) \, d\mathbf{X}^{t-1}. \end{aligned} \tag{8}$$

The closed-form solution of (8) is generally not possible so it is necessary to use an approximation. The development of a tractable particle filtering approximation is the subject of this paper. Before proceeding with a description of the proposed algorithm, a review of particle filtering is presented. This material is covered in greater depth in [18].

A. Approximation of the JMPD via Particle Filtering

In particle filtering using sequential importance sampling a set of samples from the JMPD $p(\boldsymbol{\xi}^0, \ldots, \boldsymbol{\xi}^t \mid \mathbf{Z}^t)$ is obtained by drawing from an importance density $q(\cdot \mid \mathbf{Z}^t)$. The JMPD is then represented by $\{\boldsymbol{\xi}_k^0, \ldots, \boldsymbol{\xi}_k^t, w_k^t\}_{k=1}^n$, where $\boldsymbol{\xi}_k^0, \ldots, \boldsymbol{\xi}_k^t$ is a sample drawn from q, w_k^t is the ratio of the JMPD to the importance density evaluated at the kth sample, and n is the sample size. The collection of weighted samples $\{\boldsymbol{\xi}_k^0, \ldots, \boldsymbol{\xi}_k^t, w_k^t\}_{k=1}^n$ will be referred to throughout the paper as an approximation, or more specifically as the PF approximation, to the JMPD. The word approximation is used in the sense that this is a finite sample size approximation, which becomes exact as the sample size approaches infinity, provided certain regularity conditions are met [32]. Note that each particle proposes the existence of a number of targets, denoted r_k^t for the kth particle at time t, in addition to the states for each of those targets, collected into $\mathbf{X}_k^t$ for the kth particle at time t. A sequential procedure is obtained by selecting the importance density to factorize as [18]

$$\begin{aligned} q(\boldsymbol{\xi}^0, \ldots, \boldsymbol{\xi}^t \mid \mathbf{Z}^t) &= q(\boldsymbol{\xi}^t \mid \boldsymbol{\xi}^0, \ldots, \boldsymbol{\xi}^{t-1}, \mathbf{Z}^t) \\ &\quad \times q(\boldsymbol{\xi}^0, \ldots, \boldsymbol{\xi}^{t-1} \mid \mathbf{Z}^{t-1}). \end{aligned} \tag{9}$$

The factorization (9) means that, given $\{\boldsymbol{\xi}_k^0, \ldots, \boldsymbol{\xi}_k^{t-1}, w_k^{t-1}\}_{k=1}^n$, the PF approximation to the JMPD at time t can be found by performing the following steps for $k = 1, \ldots, n$:

$$\boldsymbol{\xi}_k^t \sim q\left(\cdot \mid \boldsymbol{\xi}_k^0, \ldots, \boldsymbol{\xi}_k^{t-1}, \mathbf{Z}^t\right) \tag{10}$$

$$w_k^t \propto w_k^{t-1} \frac{p\left(\mathbf{z}^t \mid \boldsymbol{\xi}_k^t, \mathbf{Z}^{t-1}\right) p\left(\boldsymbol{\xi}_k^t \mid \boldsymbol{\xi}_k^{t-1}\right)}{q\left(\boldsymbol{\xi}_k^t \mid \boldsymbol{\xi}_k^0, \ldots, \boldsymbol{\xi}_k^{t-1}, \mathbf{Z}^t\right)}. \tag{11}$$

In order to ensure that particles remain approximately evenly weighted, a necessity for accurate approximation, the particle set should be resampled at regular intervals [33]. Particles are selected for inclusion in the resampled particle set based on their weighting with highly weighted particles selected many times

and lowly weighted particles selected rarely. The filtering density $p(\boldsymbol{\xi}^t \mid \mathbf{Z}^t)$, of most interest in target tracking, is represented by $\{\boldsymbol{\xi}_k^t, w_k^t\}_{k=1}^n$.

IV. TARGET TRACKING FOR A FIXED AND KNOWN NUMBER OF TARGETS

The problem considered in this section is that of tracking a known and fixed number of targets. Prior distributions for the state of each target are assumed to be available. The PF designed to solve this problem will form the basis of our solution to the more interesting problem in which the number of targets is unknown and varies in time. The fixed and known number of targets will be denoted as r in this section.

A. Improvement Strategies

It is relatively straightforward to develop a PF which, given a sufficient sample size, will provide an accurate approximation to the JMPD. However a solution of minimal computational expense requires careful design of the PF by using the inherent structure of the tracking model to reduce the amount of numerical simulation the PF is required to perform. This aim can be achieved by designing an importance density with the following features.

1) *Joint measurement-directed proposals*: The importance density should propose particles jointly for all target states conditional upon all previous target states and the current measurement. In [16] and [17], particles for each target were proposed indepedently leading to many particles being proposed in undesirable parts of the multitarget state space when multiple targets are in close proximity. A joint proposal overcomes this by taking into account the presence of nearby targets. An importance density which is conditional on the current measurement is commonly referred to as the optimal importance density (OID) [22]. Here, we use the terminology joint OID (JOID) to emphasize that target states are drawn jointly.

2) *Target clustering*: The computational expense of jointly drawing samples for a group of targets increases exponentially with the number of targets. An increase in the number of target states being proposed jointly is also accompanied by performance degradation for smaller sample sizes. Fortunately, the structure of the MTT problem is such that joint sampling is required only for targets in close proximity. Thus, the targets can be separated into $s \leq r$ clusters $C_1, \ldots, C_s$ such that $\bigcup_{l=1}^s C_l = \{1, \ldots, r\}$ and $\forall l \in \{1, \ldots, s\}, \forall i \in C_l$

$$\left| \hat{\boldsymbol{\rho}}_i^{t \mid t-1} - \hat{\boldsymbol{\rho}}_j^{t \mid t-1} \right| \leq \Gamma \Rightarrow j \in C_l \tag{12}$$

where $\hat{\boldsymbol{\rho}}_i^{t \mid t-1}$ is the predicted position of the ith target and Γ is a threshold. The positions of the targets in the lth cluster are collected into $\mathbf{c}_l^t$. For a sufficiently large value of Γ, the likelihood satisfies

$$p(\mathbf{z}^t \mid \mathbf{X}^t) \approx \kappa \prod_{l=1}^s \pi \left(\mathbf{z}^t \mid \mathbf{c}_l^t \right) \tag{13}$$

where κ is a constant independent of $\mathbf{X}^t$, and $\pi(\mathbf{z} \mid \mathbf{c})$ is the likelihood of the measurement vector $\mathbf{z}$ conditional on the target cluster $\mathbf{c}$ under the assumption that only this target cluster exists. The π notation will be used throughout this paper to indicate densities evaluated in this manner. The approximate factorization of (13) is exploited in the sampling procedure by drawing samples for each cluster independently. The samples for each cluster are then combined and weighted to account for the approximation involved in assuming this factorization. This approach has been used in [16], [17], and [24], where the primary motivation for separating the targets into clusters was to improve performance. When the JOID is used for each cluster the degree of performance improvement gained by clustering, while important, is perhaps of less significance than the computational savings.

3) *Rao–Blackwellization*: Nonlinearity in the tracking model enters in the relationship between the measurements and the position elements of the target state. Since the measurements are generated independently of the target velocity and the relationship between the velocity elements and position elements of the target state is linear and Gaussian, there is no need to include the velocity elements in the PF. Conditional upon a trajectory of the position elements, which are obtained from a particle filter, the posterior distribution of the velocity elements can be computed using a Kalman filter. This idea has been used, for instance, in maneuvering target tracking [34].

B. Development of the Tracking Algorithm

An importance density employing the three refinements described above will now be developed. Since the target number is assumed known and fixed in this section it is not included in the state vector so that $\boldsymbol{\xi}^t = \mathbf{X}^t$. Let $\boldsymbol{\rho}_i^t$ denote the position elements and $\mathbf{v}_i^t$ the velocity elements of the ith target, $i = 1, \ldots, r$. The dynamic equation (2) for the ith target can then be expanded as

$$\boldsymbol{\rho}_i^t = \boldsymbol{\rho}_i^{t-1} + T \mathbf{v}_i^{t-1} + \boldsymbol{\epsilon}_i^t \tag{14}$$

$$\mathbf{v}_i^t = \mathbf{v}_i^{t-1} + \boldsymbol{\eta}_i^t \tag{15}$$

where $(\boldsymbol{\epsilon}_i^{t\prime}, \boldsymbol{\eta}_i^{t\prime})' \sim N(\mathbf{0}, \mathbf{Q}_i^t)$. Since (14) and (15) form a linear/Gaussian stochastic dynamic system with (15) the "process" equation and (14) the "measurement" equation, the distribution of $\mathbf{v}_i^t$ conditional upon the position trajectory $\boldsymbol{\rho}_i^0, \ldots, \boldsymbol{\rho}_i^t$ is Gaussian and can be found exactly using the Kalman filter (KF). Equations (14) and (5) form a nonlinear/non-Gaussian stochastic dynamic system in which the distribution of $\mathbf{P}^t$ conditional on $\mathbf{Z}^t$ cannot be found exactly but instead must be approximated. This suggests decomposing the JMPD of $\mathbf{X}^0, \ldots, \mathbf{X}^t$ as

$$p\left(\mathbf{X}^0, \ldots, \mathbf{X}^t \mid \mathbf{Z}^t\right) = p\left(\mathbf{P}^0, \ldots, \mathbf{P}^t \mid \mathbf{Z}^t\right)$$
$$\times \prod_{i=1}^r p\left(\mathbf{v}_i^0, \ldots, \mathbf{v}_i^t \mid \boldsymbol{\rho}_i^0, \ldots, \boldsymbol{\rho}_i^t, \mathbf{Z}^t\right). \tag{16}$$

The densities $p(\mathbf{v}_i^0, \ldots, \mathbf{v}_i^t \mid \boldsymbol{\rho}_i^0, \ldots, \boldsymbol{\rho}_i^t, \mathbf{Z}^t), i = 1, \ldots, r$ can be computed using the KF and $p(\mathbf{P}^0, \ldots, \mathbf{P}^t \mid \mathbf{Z}^t)$ can be approximated using a PF. The posterior density of the velocity elements

can be computed using well-known recursions after allowing for the dependence between ϵ_i^t and η_i^t [35]. The details are omitted for the sake of brevity.

The PF approximation to the JMPD of the target positions at time $t-1$ is represented by the particle set $\{\mathbf{P}_k^{t-1}, w_k^{t-1}\}_{k=1}^n$. Note that $w_k^{t-1} = 1/n, k = 1, \ldots, n$ for reasons that will be explained below. With these weightings, the particle set can be considered to approximate the JMPD as

$$\hat{p}(\mathbf{P}^{t-1} \,|\, \mathbf{Z}^{t-1}) = 1/n \sum_{k=1}^n \Delta\left(\mathbf{P}^{t-1}, \mathbf{P}_k^{t-1}\right) \qquad (17)$$

where, for $\mathbf{y}_i \in \mathbb{R}^{d_i}, i = 1, 2$,

$$\Delta(\mathbf{y}_1, \mathbf{y}_2) = \begin{cases} \delta(\mathbf{y}_1 - \mathbf{y}_2), & d_1 = d_2 \\ 0, & \text{otherwise.} \end{cases} \qquad (18)$$

with δ denoting Dirac's delta function. In this section, where the number of targets is known, Δ reduces to Dirac's delta in (17). The PF approximation to the JMPD at time t can then be found as

$$\hat{p}(\mathbf{P}^t \,|\, \mathbf{Z}^t) \propto p(\mathbf{z}^t \,|\, \mathbf{P}^t)/n \sum_{k=1}^n p\left(\mathbf{P}^t \,\middle|\, \mathbf{P}_k^{t-1}\right). \qquad (19)$$

The PF seeks a set of samples from (19). In the case of MTT, it is desired to increase the efficiency of the sampling process by separating the targets into s clusters based on the predicted position of each target. Using the notation established previously, the indices of the targets in the lth cluster are contained in C_l. The kth particle $\mathbf{P}_k^{t-1}$ is composed of s clusters $\mathbf{c}_{1,k}^{t-1}, \ldots, \mathbf{c}_{s,k}^{t-1}$. The basic idea is to construct particles at time t from target clusters that belonged to different particles at time $t-1$. This is suggested by the factorization (13) and results in a significant decrease in computational expense, particularly if the number of targets in the largest cluster is much smaller than the total number of targets. Since (13) is approximate a weighting adjustment is required when clusters from different particles are combined to form the reconstructed particles. This weighting adjustment is derived below.

The process of constructing particles by collecting clusters from several particles is equivalent to sampling from

$$p\left(\mathbf{c}_1^t, d_1, \ldots, \mathbf{c}_s^t, d_s \,\middle|\, \mathbf{Z}^t, \mathcal{P}^{t-1}\right) \propto p(\mathbf{z}^t \,|\, \mathbf{P}^t) \prod_{l=1}^s p\left(\mathbf{c}_l^t \,\middle|\, \mathbf{c}_{l,d_l}^{t-1}\right) \qquad (20)$$

where $d_l \in \{1, \ldots, n\}, l = 1, \ldots, s$ is the index of the particle from which the lth cluster will be selected and $\mathcal{P}^{t-1} = \{\mathbf{P}_k^{t-1}\}_{k=1}^n$ is the collection of particle positions at time $t-1$. This is similar to the formulation of the auxiliary PF in which an auxiliary variable is used to aid selection of the best particles [36].

Sampling from (20) is performed through an importance density, which factorizes as

$$q\left(\mathbf{c}_1^t, d_1, \ldots, \mathbf{c}_s^t, d_s \,\middle|\, \mathbf{Z}^t, \mathcal{P}^{t-1}\right) = \prod_{l=1}^s q\left(\mathbf{c}_l^t, d_l \,\middle|\, \mathbf{Z}^t, \mathcal{C}_l^{t-1}\right) \qquad (21)$$

where $\mathcal{C}_l^{t-1} = \{\mathbf{c}_{l,k}^{t-1}\}_{k=1}^n$. The importance density for the lth cluster is taken to be

$$q\left(\mathbf{c}_l^t, d_l \,\middle|\, \mathbf{Z}^t, \mathcal{C}_l^{t-1}\right) = \psi_{l,d_l} \pi\left(\mathbf{c}_l^t \,\middle|\, \mathbf{c}_{l,d_l}^0, \ldots, \mathbf{c}_{l,d_l}^{t-1}, \mathbf{z}^t\right) \qquad (22)$$

where $\psi_{l,k}$ is the probability of selecting the lth cluster from the kth particle. The trajectory $\mathbf{c}_l^0, \ldots, \mathbf{c}_l^{t-1}$ should be interpreted to mean the collection of trajectories of positions in the cluster C_l at time t. A draw from (22) consists of selecting a particle index from $\{1, \ldots, n\}$ according to the weights $\psi_{l,1}, \ldots, \psi_{l,n}$ and then drawing from the JOID conditional upon the position at time $t-1$ of the selected particle. The weights $\psi_{l,1}, \ldots, \psi_{l,n}$ can be chosen arbitrarily although proper setting of these weights is necessary to ensure that the best clusters are selected for inclusion in the reconstructed particles. The most sensible choice seems to be the usual OID weight update [22]

$$\psi_{l,k} \propto \pi\left(\mathbf{z}^t \,\middle|\, \mathbf{c}_{l,k}^0, \ldots, \mathbf{c}_{l,k}^{t-1}\right), \quad k = 1, \ldots, n. \qquad (23)$$

The benefit of using (23) will become clear when the weight update for the reconstructed particle is considered.

In the following, working expressions for the JOID and the weights (23) are derived. Assume a cluster of h targets and let $\mathbf{c}^t = (\boldsymbol{\rho}_1^{t\prime}, \ldots, \boldsymbol{\rho}_h^{t\prime})'$ denote the collection of target positions and $\mathbf{c}^0, \ldots, \mathbf{c}^{t-1}$ denote a trajectory of positions for these targets from time 0 to time $t-1$. Using Bayes' rule, the JOID can be written as

$$\pi(\mathbf{c}^t \,|\, \mathbf{c}^0, \ldots, \mathbf{c}^{t-1}, \mathbf{z}^t) = \frac{\pi(\mathbf{z}^t \,|\, \mathbf{c}^t) p(\mathbf{c}^t \,|\, \mathbf{c}^0, \ldots, \mathbf{c}^{t-1})}{\pi(\mathbf{z}^t \,|\, \mathbf{c}^0, \ldots, \mathbf{c}^{t-1})}. \qquad (24)$$

Let $\hat{\mathbf{v}}_i^{t-1\,|\,t-1}$ and $\boldsymbol{\Sigma}_i^{t-1\,|\,t-1}, i = 1, \ldots, h$ denote the posterior mean and covariance matrix of the velocity of the ith target conditional on the position trajectory. The transition density for the target positions can be found as

$$p(\mathbf{c}^t \,|\, \mathbf{c}^0, \ldots, \mathbf{c}^{t-1}, \mathbf{Z}^{t-1}) = \prod_{i=1}^h N\left(\boldsymbol{\rho}_i^t; \hat{\boldsymbol{\rho}}_i^t, \boldsymbol{\Psi}_i^t\right) \qquad (25)$$

where $\hat{\boldsymbol{\rho}}_i^t = \boldsymbol{\rho}_i^{t-1} + T\hat{\mathbf{v}}_i^{t-1\,|\,t-1}$ and $\boldsymbol{\Psi}_i^t = T^2 \boldsymbol{\Sigma}_i^{t-1\,|\,t-1} + \mathbf{Q}_{i,\rho}^t$ with $\mathbf{Q}_{i,\rho}^t$ the process noise covariance matrix for the position elements.

Since the value returned by a particular cell depends directly on the number of targets occupying that cell, the likelihood can be found by expanding over all possible occupation configurations. Let $M_i, i = 1, \ldots, h$ denote measurement cells in the neighborhood of the position of the ith target. The neighborhood can be thought of as the collection of cells which the target could possibly occupy. Let $V_j, j = 1, \ldots, J$ denote the region of measurement space occupied by the jth measurement cell. These

Fig. 1. A measurement grid of $J = 20$ cells for a two-dimensional surveillance region occupied by one target. The region of space occupied by cell 2 is indicated by the solid shaded area, denoted as V_2. The regions occupied by the other measurement cells are similarly labeled. The predicted target position is indicated by a cross with the neighborhood of the target indicated by the diagonally shaded area. The indices of the cells forming the neighborhood are collected into $M_1 = \{13, 14, 18, 19\}$. Note that the neighborhood can potentially contain several cells due to uncertainty in the predicted target position, as determined by the process noise and the uncertainty in the velocity elements.

quantities are depicted in Fig. 1 for a two-dimensional surveillance region. The likelihood conditional on the target cluster can be expanded as

$$\pi(\mathbf{z}^t \,|\, \mathbf{c}^t) = \prod_{j=1}^{J} l_0\left(z_j^t\right) \sum_{j_1 \in M_1} \cdots \sum_{j_h \in M_h} \omega_{j_1,\ldots,j_h} \prod_{i=1}^{h} \chi_{V_{j_i}}\left(\boldsymbol{\rho}_i^t\right) \tag{26}$$

where $\chi_A(z) = 1$ if $z \in A$ and zero otherwise and

$$\omega_{j_1,\ldots,j_h} = \prod_{u=1}^{\bar{h}} \frac{l_{m_u}\left(z_{\bar{j}_u}^t\right)}{l_0\left(z_{\bar{j}_u}^t\right)} \tag{27}$$

with $\bar{h}$ the number of distinct cell indices in $\{j_1, \ldots, j_h\}$, $\bar{j}_1, \ldots, \bar{j}_{\bar{h}}$ the distinct cell indices, and $m_1, \ldots, m_{\bar{h}}$ the multiplicities of the distinct cells. For the Rayliegh-distributed measurement model used here the cell measurement likelihood l_m is given by (6) for nonthresholded measurements and (7) for thresholded measurements. The JOID can be found by substituting (25) and (26) into (24) to obtain

$$\pi(\mathbf{c}^t \,|\, \mathbf{c}^0, \ldots, \mathbf{c}^{t-1}, \mathbf{Z}^t)$$
$$= \sum_{j_1 \in M_1} \cdots \sum_{j_h \in M_h} \beta_{j_1,\ldots,j_h} \phi_{j_1,\ldots,j_h}(\mathbf{c}^t) \tag{28}$$

where

$$\beta_{j_1,\ldots,j_h} = \alpha_{j_1,\ldots,j_h} \bigg/ \sum_{e_1 \in M_1} \cdots \sum_{e_h \in M_h} \alpha_{e_1,\ldots,e_h} \tag{29}$$

$$\alpha_{j_1,\ldots,j_h} = \omega_{j_1,\ldots,j_h} \prod_{i=1}^{h} \gamma_{i,j_i} \tag{30}$$

$$\phi_{j_1,\ldots,j_h}(\mathbf{c}^t) = \prod_{i=1}^{h} \left\{ \chi_{V_{j_i}}\left(\boldsymbol{\rho}_i^t\right) N\left(\boldsymbol{\rho}_i^t; \hat{\boldsymbol{\rho}}_i^t, \boldsymbol{\Psi}_i^t\right) \big/ \gamma_{i,j_i} \right\} \tag{31}$$

with

$$\gamma_{i,j} = \int_{V_j} N\left(\boldsymbol{\rho}; \hat{\boldsymbol{\rho}}_i^t, \boldsymbol{\Psi}_i^t\right) d\boldsymbol{\rho}. \tag{32}$$

It can be seen from (28) that the JOID is a mixture of truncated Gaussian distributions. Each component can be interpreted as a hypothesis on the cell locations of the q targets in the cluster. The components are weighted according to the chances of each target moving to its assigned cell as determined by the received measurements and the assumed model for the target dynamics. A draw from (28) can be made by selecting a mixture component using the probabilities of (29) and then drawing each target position from the appropriate truncated Gaussian distribution.

In this paper, the collection of cells forming the neighbourhood of the ith target is defined as

$$M_i = \{j \in \{1, \ldots, J\} : \gamma_{i,j} > \Upsilon\} \tag{33}$$

where Υ is a small, predefined lower bound. The number of cells in M_i, and hence the computational expense, will obviously decrease as Υ is increased. However, feasible target destinations may be ignored if Υ is too large. In this paper, $\Upsilon = 0.005$ is used as an acceptable compromise between these conflicing requirements.

The weights (23) are given by the normalization factor for the JOID, which can be found as

$$\pi(\mathbf{z} \,|\, \mathbf{c}^0, \ldots, \mathbf{c}^{t-1}) = \int \pi(\mathbf{z} \,|\, \mathbf{c}^t) p(\mathbf{c}^t \,|\, \mathbf{c}^0, \ldots, \mathbf{c}^{t-1}) \, d\mathbf{c}^t$$
$$\propto \sum_{j_1 \in M_1} \cdots \sum_{j_q \in M_q} \alpha_{j_1,\ldots,j_q}. \tag{34}$$

It remains to compute the weight adjustment for the reconstructed particles. Since reconstructed particles contain target clusters originating from different particles the weight of a particle at time $t - 1$ will have no connection to the reconstructed particle at time t. The reconstructed particles must therefore be resampled at each time-step so that $w_k^t = 1/n$ for $k = 1, \ldots, n, t = 0, 1, \ldots$. Let $\mathbf{c}_{l,1}^t, \ldots, \mathbf{c}_{l,n}^t$ denote the particles drawn for the lth target cluster. The weight of the kth particle can be found, by substituting (21), (22), (23), and (20) into (11), as

$$\tilde{w}_k^t \propto p\left(\mathbf{z}^t \,|\, \mathbf{c}_{1,k}^t, \ldots, \mathbf{c}_{s,k}^t\right) \bigg/ \prod_{l=1}^{s} \pi\left(\mathbf{z}^t \,|\, \mathbf{c}_{l,k}^t\right). \tag{35}$$

Resampling based on $\tilde{w}_1^t, \ldots, \tilde{w}_n^t$ is performed in order to obtain an evenly weighted particle set, i.e., $w_k^t = 1/n$ for $k = 1, \ldots, n$. It follows from (13) that for a sufficiently large clustering threshold Γ, a large number of distinct particles will be selected so that particle duplication is minimized. Equation (35) also motivates the choice of weighting given in (23) since any other choice will not have the property that uniform weights are obtained for a sufficiently large clustering threshold. This implies that other weighting choices do not select the best clusters for reconstruction.

The joint likelihoods required in (35) can be computed using (26) with $M_i, i = 1, \ldots, r$ a singleton containing the cell occupied by the ith target at time t. Note that the computational

expense of computing the joint likelihood does not increase exponentially with the number of targets as it is only necessary to consider the given target positions at time t. This is not true of the sampling procedure, as can be seen from (28), because there it is necessary to consider possible target locations at time t given the target locations at time $t - 1$.

C. Selecting the Clustering Threshold

The performance of an importance sampling scheme is heavily influenced by the variance of the weights with a low variance required for good performance [37]. In the MTT filter described above there are two sets of weights to consider: the first-stage weights defined in (23) and second-stage weights of (35). Under given conditions, the variances of these two sets of weights are determined by the clustering threshold. In particular, the first-stage weights have a low variance when the clustering threshold is small so that only small numbers of targets are sampled jointly. Conversely, the second-stage weights will have a low variance when the clustering threshold is large so that the factorization (13) is satisfied. In this section, the variances of the first- and second-stage weights are examined with a view to determining a clustering threshold that balances these conflicting requirements. The relevant derivations have been omitted for the sake of brevity but can be found in [38].

The weights variances are examined for a scenario containing ten targets moving in a plane at a time at which there are several targets in close proximity. The configuration of the targets is given in Fig. 2(a). This is a snapshot taken at time 648 of the scenario described in Section IV-D. The SNR is 6 dB, measurements are thresholded to give a false alarm probability of 0.01 and the grid size is 100 m × 100 m. Particles for each target are distributed in a small region about the true target position. The sample size is 50 particles. For a clustering threshold that separates the targets into s clusters, there will be s first-stage weight variances. The product of the first-stage weights variances is used as an overall measure of the first-stage weights variances and will be referred to as the first-stage weight variance for convenience. In the top part of Fig. 2(b), the weight variances are plotted on a log scale against clustering threshold with the first- and second-stage weight variances shown as solid and dashed lines, respectively. Shown on the lower part of the same figure is a measure of tracking performance plotted against clustering threshold. Tracking performance is measured by the mean number of targets in track throughout the surveillance period, as described in Section IV-D. This is computed for each clustering threshold over 100 realizations.

Initially, for $\Gamma < 90$, no clustering is performed so that targets are processed individually. This is seen to result in a negligible first-stage weight variance and a very large second-stage weight variance. As the clustering threshold increases, the factorization (13) becomes increasingly valid and the second-stage weight variance decreases. At the same time the cluster sizes increase causing an increase in the first-stage weight variance. A final, large decrease in the second-stage weight variance is obtained for $\Gamma > 240$. This decrease corresponds to the addition of target 6 to a cluster containing targets 3, 4, and 5. Further increases in the clustering threshold will cause an increase in the first-stage
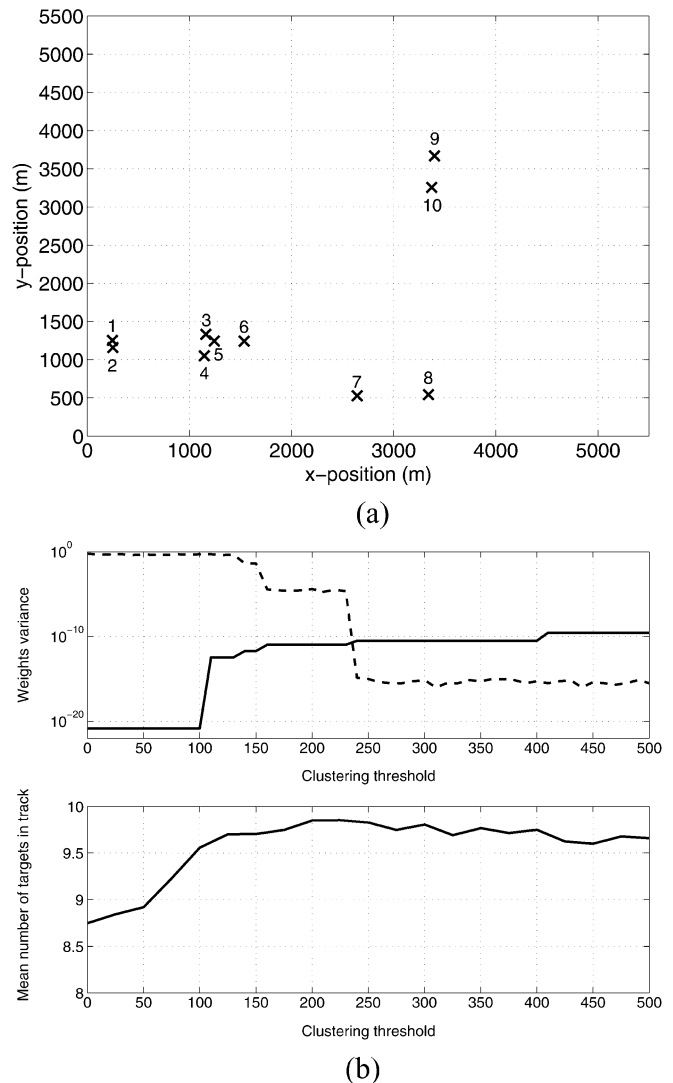

Fig. 2. Ten target scenario for examining the effect of clustering on performance. (a) Configuration of targets at time 648. (b) Top plot shows the variances of the first-stage (solid line) and second-stage (dashed line) weights at time 648 plotted against clustering threshold. The bottom plot shows the mean number of targets tracked plotted against clustering threshold.

weight variance with no compensating decrease in second-stage weight variance.

The link between weights variance and performance is clearly evident from the bottom part of Fig. 2. Performance is largely determined by the second-stage weights variance although increases in the first-stage weights variance for large clustering thresholds do have some impact on performance. The best performance is achieved for clustering thresholds in the range $200 \leq \Gamma \leq 300$. A clustering threshold in this range provides excellent performance with only a small increase in computational expense compared to a threshold of zero. Note that the performance obtained for a clustering threshold of zero is that achieved by a procedure that draws samples independently for each target, as in [17]. The significant performance benefits of joint sampling over independent sampling are clearly evident.
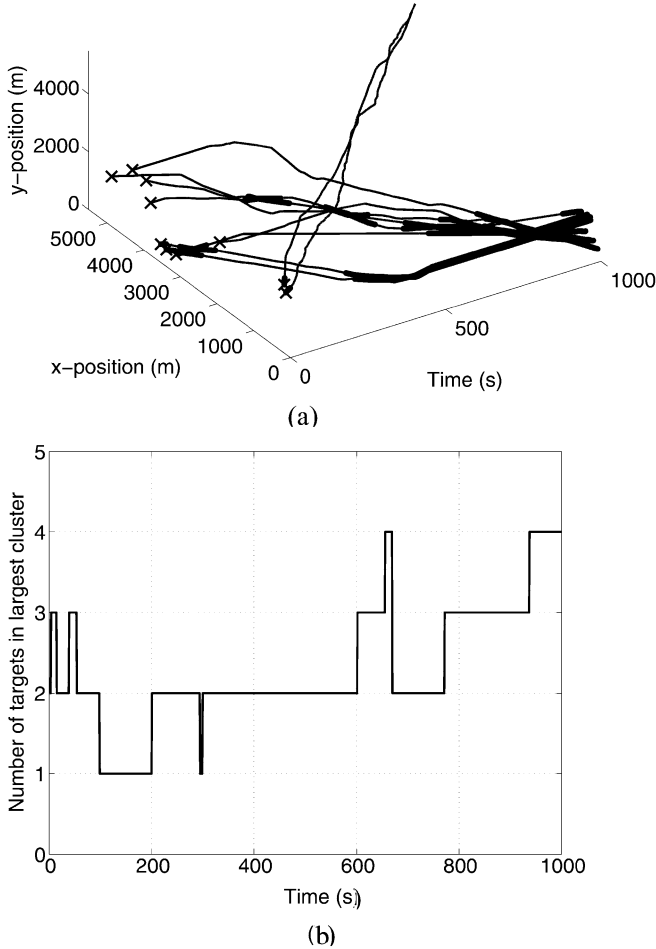
Fig. 3. Simulation scenario: (a) target trajectories with positions at time zero indicated by crosses and periods in which multiple targets are in close proximity highlighted and (b) the largest cluster size for the true target positions plotted against time for a clustering threshold of 250.

D. Simulations

The scenario considered here involves ten targets moving in a two-dimensional region of size 5500 m × 5500 m for 1000 time-steps of 1 s each. The target trajectories belong to ground vehicles involved in an exercise at the U.S. Army's National Training Center. A plot showing the target trajectories for the entire surveillance period is given in Fig. 3(a). Target positions at time zero are indicated by crosses and time-steps at which multiple targets are in close proximity are highlighted. The observation region is divided into 100 m × 100 m cells with measurements generated from the target trajectories according to the thresholded measurement model described in Section II. Monte Carlo realizations are obtained by generating independent measurement sequences from the same set of target trajectories. The filter uses the dynamic model (2) with process noise covariance matrix

$$\mathbf{Q}_i^t = 20 \begin{pmatrix} 1 & 0 \\ 0 & 1/100 \end{pmatrix} \otimes \mathbf{I}_2. \tag{36}$$

This process noise attempts to accommodate the actual target behavior which contains periods of constant velocity motion interspersed with sudden accelerations and move–stop–move be-

havior. The initial distribution for the ith target is $N(\mathbf{x}_i^0, \mathbf{P}^0)$, where $\mathbf{x}_i^0$ is the true state of the ith target at time zero and

$$\mathbf{P}^0 = \begin{pmatrix} 225 & 0 \\ 0 & 4 \end{pmatrix} \otimes \mathbf{I}_2. \tag{37}$$

The sampling procedure of Section IV-B is applied with a clustering threshold of 250 m. Since the complexity of a multiple target tracking problem generally increases as the number of targets in close proximity increases it is of interest to examine the degree of closeness implied by this clustering threshold. Based on the true target positions, the largest cluster size at each time-step using a clustering threshold of 250 m is plotted in Fig. 3(b). It can be seen that multiple targets are in close proximity for much of the time with as many as four targets forming a cluster. The cluster sizes used by the filter may differ since clustering is performed based on estimated target positions.

The reliability and tracking accuracy of the algorithm are measured in the performance analysis. Assume that m Monte Carlo measurement sequences are generated from r target trajectories of length l time-steps to assess algorithm performance. Let $\hat{\boldsymbol{\rho}}_i^t(j), i = 1, \ldots, r, t = 1, \ldots, l, j = 1, \ldots, m$ denote the position estimate of the ith target on the jth realization at time t. Reliability is measured by the mean number of targets in track averaged over all time-steps and all realizations. This statistic is computed as

$$\hat{r} = \frac{1}{lm} \sum_{i=1}^{r} \sum_{t=1}^{l} \sum_{j=1}^{m} \chi_{[0,\Lambda]}\left(\left\|\boldsymbol{\rho}_i^t - \hat{\boldsymbol{\rho}}_i^t(j)\right\|_2\right) \tag{38}$$

where Λ is a threshold. The root mean square (RMS) position error averaged over targets that are deemed to be in track at each time-step is also computed. The idea here is to capture the tracking accuracy by considering only those targets that the filter is actually following. Let $T_{i,j} = \{t \in \{1, \ldots, l\} : \|\|\boldsymbol{\rho}_i^t - \hat{\boldsymbol{\rho}}_i^t(j)\|_2 \in [0, \Lambda]\}$ denote the set of time-steps at which the ith target is in track on the jth realization. The tracking accuracy is then measured by

$$E = \frac{1}{rm} \sum_{i=1}^{r} \sum_{j=1}^{m} \frac{1}{|T_{i,j}|} \sum_{t \in T_{i,j}} \left\|\boldsymbol{\rho}_i^t - \hat{\boldsymbol{\rho}}_i^t(j)\right\|_2. \tag{39}$$

Good tracking performance requires both reliability, i.e., the filter should not "lose" the targets, and accuracy, i.e., filter estimates of the targets being tracked should be close to the true values.

In the following analysis $m = 100$ realizations are used for several values of the SNR and detection threshold. The detection threshold is characterized here by the probability of registering a detection in a cell occupied by one target. This will be referred to as the detection probability and denoted as P_D although it should be kept in mind that it is actually the single target detection probability. The results are given in Tables I and II for SNRs of $10 \log_{10}(\lambda) = 10$ and 5 dB, respectively. The false alarm probabilities for each case are computed using (7) as $P_{\text{FA}} = P_D^{1+\lambda}$. The threshold Λ is set to 150 m. The results indicate that good tracking performance, in terms of both reliability and accuracy, is achieved for both SNRs considered here. Importantly, this performance is achieved using sample sizes of no more than 500 particles. This implies a computational

TABLE I
PERFORMANCE OF THE PROPOSED ALGORITHM AT AN SNR OF 10 dB

Mean number of targets in track.						RMS position error for targets in track					
	Number of particles						Number of particles				
$P_D(P_{FA})$	25	50	100	200	500	$P_D(P_{FA})$	25	50	100	200	500
$0.3(1.8 \times 10^{-6})$	9.87	9.87	9.92	9.94	9.96	$0.3(1.8 \times 10^{-6})$	39.38	38.36	37.88	37.66	37.54
$0.4(4.2 \times 10^{-5})$	9.95	9.99	9.99	10	10	$0.4(4.2 \times 10^{-5})$	37.74	36.95	36.52	36.35	36.22
$0.5(4.9 \times 10^{-4})$	9.98	9.98	10	10	10	$0.5(4.9 \times 10^{-4})$	36.95	36.43	36.10	35.97	35.89
$0.6(3.6 \times 10^{-3})$	9.94	9.98	10	10	10	$0.6(3.6 \times 10^{-3})$	37.01	36.50	36.25	36.16	36.08
$0.7(0.020)$	9.91	9.99	9.99	10	10	$0.7(0.020)$	37.53	37.09	36.87	36.75	36.70

TABLE II
PERFORMANCE OF THE PROPOSED ALGORITHM AT AN SNR OF 5 dB

Mean number of targets in track.						RMS position error for targets in track					
	Number of particles						Number of particles				
$P_D(P_{FA})$	25	50	100	200	500	$P_D(P_{FA})$	25	50	100	200	500
$0.3(6.7 \times 10^{-3})$	9.12	9.58	9.80	9.87	9.91	$0.3(6.7 \times 10^{-3})$	42.90	41.77	41.20	40.96	40.70
$0.4(0.022)$	8.98	9.55	9.76	9.89	9.95	$0.4(0.022)$	42.80	41.64	41.16	40.80	40.58
$0.5(0.056)$	8.71	9.55	9.76	9.85	9.94	$0.5(0.056)$	43.07	41.89	41.28	40.95	40.72
$0.6(0.119)$	8.49	9.28	9.70	9.82	9.92	$0.6(0.119)$	44.21	42.89	42.02	41.61	41.31
$0.7(0.227)$	7.82	8.80	9.49	9.65	9.83	$0.7(0.227)$	46.31	44.69	43.88	43.27	42.84

tractability which would permit real-time tracking of multiple targets. For instance, with a sample size of 500 particles, the C implementation of the algorithm takes 0.18 s to process a set of measurements on a Pentium IV with a 2.4-GHz processor. Note that although reasonable accuracy is obtained with small sample sizes, the reliability of the filter is significantly worse than that obtained with larger sample sizes, particularly for the lower SNR.

It can be seen that, provided the filter has a target in track, the accuracy is only marginally affected by SNR, detection threshold or even sample size. The factor most influencing performance is then the ability of the filter to maintain track on a target, referred to here as reliability. As expected, a decrease in SNR causes the reliability of the algorithm to deteriorate. This decline is most apparent when using smaller sample sizes. The choice of detection threshold appears to depend on both the SNR and the sample size. The optimum detection threshold, in the sense that it maximizes the reliability, increases as the SNR increases. The degree to which the optimum detection threshold depends on SNR increases as the sample size decreases. For the SNRs considered here, increases in the sample size beyond 500 provide only marginal improvements in either reliability or accuracy.

V. TARGET TRACKING AND DETECTION FOR AN UNKNOWN NUMBER OF TARGETS

In this section, MTT is considered for the general case where the number of targets is unknown and/or time varying. As discussed in Section III, the JMPD provides a framework for gen-

eral MTT. The previous section demonstrated that, for the case of a known number of targets, the JMPD can be approximated in a computationally efficient manner using a carefully designed PF. The aim of this section is to build on this design to develop an efficient procedure for approximation of the JMPD when the number of targets is unknown.

Numerical approximation of the JMPD is much more difficult when the number of targets is unknown. The state variable is now the extended multitarget state $\boldsymbol{\xi}^t = (\mathbf{X}^{t\prime}, r^t)'$, which resides in the space $\bigcup_{i=0}^{\infty}\{i\} \times \mathbb{R}^{d_x i}$ with $\mathbb{R}^0 = \emptyset$. Efficient particle proposals are essential for computationally feasible exploration of this space. The approach taken here will be to extend the idea of measurement-directed particle proposals to ensure that particles with the correct target number are proposed with high probability. Intuitively, if the target number element r^t of $\boldsymbol{\xi}^t$ is correct, then the procedure of Section IV will ensure efficient exploration of the space $\mathbb{R}^{d_x r^t}$. Before proceeding with the design of the PF, it is necessary to formulate a model for the time evolution of the extended multitarget state $\boldsymbol{\xi}^t$.

A. Development of the Transition Model

The transition density $p(\boldsymbol{\xi}^t \,|\, \boldsymbol{\xi}^{t-1})$ for the extended multitarget state can be written as

$$p(\mathbf{X}^t, r^t \,|\, \mathbf{X}^{t-1}, r^{t-1}) = p(r^t \,|\, \mathbf{X}^{t-1}, r^{t-1})$$
$$\times p(\mathbf{X}^t \,|\, r^t, \mathbf{X}^{t-1}, r^{t-1}). \quad (40)$$

The two terms in the product (40) will be considered separately in the following development. These terms are referred to as

the *target number* and the *target state* conditionals, respectively. These terms are derived under the following assumptions: a) At each time-step at most one target may arrive in each unoccupied cell, b) targets depart only from occupied cells, and c) target arrivals and departures are independent.

1) The Target Number Conditional: A change in the target number of d can be achieved through the removal of b targets and the addition of $d+b$ targets. Thus, using the total probability theorem, the target number conditional can be expanded as

$$
\begin{aligned}
\Pr(r^t &= r^{t-1} + d \,|\, \mathbf{X}^{t-1}, r^{t-1}) \\
&= \sum_b \Pr(b \text{ targets are removed}) \\
&\quad \times \Pr(d+b \text{ targets are added}).
\end{aligned}
\tag{41}
$$

The probabilities $\Pr(a \text{ targets are added})$ and $\Pr(b \text{ targets are removed})$ are determined from prior arrival and removal probabilities. Let J denote the number of cells in the measurement region. At time t, $A^t \subseteq \{1, \dots, J\}$ is the set of cells in which new targets can be added. This could be, for instance, the set of cells that are not occupied by existing targets. The number of elements of A^t will be denoted e^t. Let $\alpha_j^t, j \in A^t$ denote the prior probability of adding a target in the jth cell at time t and $\beta_i^t, i = 1, \dots, r^{t-1}$ denote the prior probability of removing the ith target at time t. These spatially varying priors can be used to encode domain specific knowledge, e.g., that targets tend to arrive or leave around the boundaries of the surveillance region.

For a set of integers Z, let

$$
\mu_m^Z(1,j), \dots, \mu_m^Z(m,j), j \in \left\{ 1, \dots, \binom{|Z|}{m} \right\}
$$

denote the jth combination of m integers from Z. Also, if $Z = \{1, \dots, z\}$, let $\mu_m^z(1,j), \dots, \mu_m^z(m,j)$ denote the jth selection of m integers from Z. Equation (41) can then be written explicitly as, for $d = -r^{t-1}, \dots, e^t$

$$
\Pr(r^t = r^{t-1} + d \,|\, \mathbf{X}^{t-1}, r^{t-1}) = \sum_{b=\max(0,-d)}^{\min(r^{t-1}, e^t - d)} \left(\sum_{j=1}^{\binom{r^{t-1}}{b}} \rho_{b,j} \right) \cdot \left(\sum_{j=1}^{\binom{e^t}{d+b}} \lambda_{d+b,j} \right)
\tag{42}
$$

where $\lambda_{a,j}$ is the probability of adding targets in the jth combination of a cells from A^t and $\rho_{b,j}$ is the probability of removing the jth combination of b targets. These probabilities are given by

$$
\lambda_{a,j} = \prod_{i \in A^t} (1 - \alpha_i^t) \prod_{l=1}^a \frac{\alpha_{\mu_a^{A^t}(l,j)}^t}{1 - \alpha_{\mu_a^{A^t}(l,j)}^t}
\tag{43}
$$

$$
\rho_{b,j} = \prod_{i=1}^{r^{t-1}} (1 - \beta_i^t) \prod_{l=1}^b \frac{\beta_{\mu_b^{r^{t-1}}(l,j)}^t}{1 - \beta_{\mu_b^{r^{t-1}}(l,j)}^t}.
\tag{44}
$$

2) The Target State Conditional: The second term in (40) is the transition for the multitarget state conditional on target number. Notionally, this can be expressed as a product of mixture densities

$$
\begin{aligned}
p(\mathbf{X}^t &\,|\, r^t = r^{t-1} + d, \mathbf{X}^{t-1}, r^{t-1}) \\
&= \sum_{b=\max(0,-d)}^{\min(r^{t-1}, e^t - d)} \sum_{\substack{\text{ways of} \\ \text{removing } b \\ \text{targets}}} \Pr(\text{removing these targets}) \\
&\quad \times p(\text{transition of remaining targets}) \\
&\quad \times \sum_{\substack{\text{ways of} \\ \text{adding } d+b \\ \text{targets}}} \Pr(\text{adding in these cells}) \\
&\quad \times p(\text{initial dist. of targets in these cells}).
\end{aligned}
\tag{45}
$$

Equation (45) allows for a degree of flexibility regarding the initial velocities and placement of new targets within cells. In the following a specific distribution for the within cell distribution of new targets is used. It is assumed that, given a new target has appeared in a particular cell, its position within that cell is uniformly distributed. Let U_A denote the density of a uniformly distributed random variable over the region A. The prior density for the velocity elements will remain unspecified and is denoted p_{v_0}. Let $U_m^z(j) = \{\mu_m^z(1,j), \dots, \mu_m^z(m,j)\}$ and for $Z \subseteq \{1, \dots, r^{t-1}\}$, let $\mathbf{X}_{-Z}^t$ be the collection of target states excluding targets with an index in Z. Equation (45) then becomes

$$
\begin{aligned}
p(\mathbf{X}^t &\,|\, r^t = r^{t-1} + d, \mathbf{X}^{t-1}, r^{t-1}) \\
&= \sum_{b=\max(0,-d)}^{\min(r^{t-1}, e^t - d)} \left(\sum_{j=1}^{\binom{r^{t-1}}{b}} \bar{\rho}_{b,j} \right. \\
&\qquad\qquad \left. \times p\left(\mathbf{X}_{-\{r^{t-1}+1-b, \dots, r^t\}}^t \,\middle|\, \mathbf{X}_{-U_b^{r^{t-1}}(j)}^{t-1} \right) \right) \\
&\quad \times \left(\sum_{j=1}^{\binom{e^t}{d+b}} \bar{\lambda}_{d+b,j} \prod_{l=1}^{d+b} U_{V_{\mu_{d+b}^{A^t}(l,j)}} \left(\boldsymbol{\rho}_{r^{t-1}+l-b}^t \right) \right. \\
&\qquad\qquad\qquad \left. \times p_{v_0}\left(\mathbf{v}_{r^{t-1}+l-b}^t \right) \right)
\end{aligned}
\tag{46}
$$

where

$$\bar{\lambda}_{a,j} = \lambda_{a,j} \Bigg/ \sum_{i=1}^{\binom{e^t}{a}} \lambda_{a,i}$$

$$\bar{\rho}_{b,j} = \rho_{b,j} \Bigg/ \sum_{i=1}^{\binom{r^{t-1}}{b}} \rho_{b,i}. \qquad (47)$$

Equation (46) is a mixture density with components weighted according to the prior arrival and removal probabilities. Comparing (45) with (46), it can be seen that, if b targets are to be removed, the probability of removing a particular combination of targets is given by $\bar{\rho}_{b,j}$ where j indexes the possible combinations. The states of persistent targets evolve according to the assumed dynamic model, as given in (2). Similarly, given that $d+b$ targets are to be added, the probability of adding the targets in a particular combination of cells is determined by $\bar{\lambda}_{d+b,j}$ where j indexes the possible combinations. For a particular combination of cells, the positions of the $d+b$ new targets are uniformly distributed in the given cells and the velocities are distributed according to p_{v_0}. The transition density for ξ^t can be found by substituting (42) and (46) into (40).

B. A Measurement-Directed Proposal Scheme

A straightforward PF implementation that draws proposals from the prior (40) will perform poorly, particularly if little prior information is encoded in the addition and removal probabilities. For instance, if the addition probabilities are uniform across the observation region sampling from (40) will result in many targets being initiated in areas of no interest. Even if reasonably useful prior information is available, measurement-directed removal and addition of targets can only enhance efficiency.

The key idea in our development of a tractable PF for integrated target detection and tracking is that of an existence grid. For each cell in the surveillance region, the existence grid provides a measure of the chances that a target is located in that cell, denoted as g_j^t in cell j at time t. The existence grid is not intended to be composed of the probabilities that a target exists in each cell. To compute such a grid is a goal of the tracking filter. The purpose of the existence grid is to act as a prop in achieving this goal. The following simplifications are used in the computation of the existence grid: a) Each cell is maintained independently of the surrounding cells. The time evolution of the existence grid therefore considers only the arrival and departure of targets in each cell according to the prior model for these events. It does not consider the movement of existing targets from one cell to another. b) The existence grid is concerned only with whether there are one or zero targets in each cell. With these simplifications the existence grid is computationally simple to maintain while providing some indication of the interesting regions of the measurement space.

Maintenance of the existence grid is performed as follows. Each cell is initialized with a prior existence probability $g_j^0, j = 1, \ldots, J$, which may vary spatially if prior information justifies.

The predicted probability of target existence in a cell is found using the total probability theorem. Expanding over the events that a target leaves or enters the cell gives, for $j = 1, \ldots, J$

$$g_j^{t|t-1} = g_j^{t-1}\left(1 - \zeta_j^t\right) + \left(1 - g_j^{t-1}\right)\epsilon_j^t \qquad (48)$$

where ϵ_j^t is the probability a new target appears in cell j at time t, and ζ_j^t is the probability a target leaves cell j at time t. When new measurements $\mathbf{z}^t$ become available, the existence grid is updated using Bayes' rule to give

$$\begin{aligned} g_j^t &= \frac{g_j^{t|t-1} p\left(\mathbf{z}^t \,|\, o_j^t = 1\right)}{p(\mathbf{z}^t)} \\ &= \frac{g_j^{t|t-1} p\left(\mathbf{z}^t \,|\, o_j^t = 1\right)}{g_j^{t|t-1} p\left(\mathbf{z}^t \,|\, o_j^t = 1\right) + (1 - g_j^{t|t-1}) p\left(\mathbf{z}^t \,|\, o_j^t = 0\right)} \end{aligned} \qquad (49)$$

where o_j^t is the number of targets occupying cell j at time t.

Under the measurement-directed proposal scheme, each particle receives a new target in cell j at a rate dictated by g_j^t rather than at the nominal rate α_j^t. Particles are used more efficiently as new targets are added only in areas deemed suitable by the measurements. Similarly, targets are removed at the rate dictated by g_j^t rather than the nominal rate given by β_t^i. This prevents unnecessary removal of targets which are supported by the measurements. This particle proposal procedure is expressed explicitly in the following importance sampling version of (42):

$$\begin{aligned} &q(r^t = r^{t-1} + d \,|\, r^{t-1}, Z^{t-1}) \\ &= \sum_{b=\max(0,-d)}^{\min(r^{t-1}, e^t - d)} \left(\sum_{j=1}^{\binom{r^{t-1}}{b}} \kappa_{b,j} \right) \left(\sum_{i=1}^{\binom{e^t}{d+b}} \nu_{d+b,i} \right), \end{aligned} \qquad (50)$$

for $d = -r^{t-1}, \ldots, e^t$, where $\nu_{a,j}$ and $\kappa_{b,j}$ are generated from the existence grid as

$$\nu_{a,j} = \prod_{i \in A^t} \left(1 - g_i^t\right) \prod_{l=1}^{a} \frac{g_{\mu_a^{A^t}(l,j)}^t}{1 - g_{\mu_a^{A^t}(l,j)}^t} \qquad (51)$$

$$\kappa_{b,j} = \prod_{i=1}^{r^{t-1}} \left(1 - \tau_i^t\right) \prod_{l=1}^{b} \frac{\tau_{\mu_b^{r^{t-1}}(l,j)}^t}{1 - \tau_{\mu_b^{r^{t-1}}(l,j)}^t} \qquad (52)$$

with

$$\tau_i^t = \beta_i^t \frac{1 - g_{v_i^t}^t}{1/r^{t-1} \sum_{l=1}^{r^{t-1}} \left(1 - g_{v_l^t}^t\right)} \qquad (53)$$

where $v_i^t \in \{1, \ldots, J\}, i = 1, \ldots, r^{t-1}$ denotes the cell occupied by the ith target. Note that the importance density for the target number, (50), is structurally identical to the transition density for the target number (42). However, the importance density uses the existence probabilities rather than the prior arrival and removal probabilities to weight the transitions in target number. This permits the measurements to influence sampling of the target number.

For $Z \subseteq \{1, \ldots, r^{t-1}\}$, let $\mathbf{P}^t_{-Z}$ be the collection of target positions excluding targets with an index in Z. The importance density for the target positions conditional on target number can be found by analogy with (46) as

$$
q(\mathbf{P}^t \mid r^t = r^{t-1} + d, \mathbf{P}^{t-1}, r^{t-1}, \mathbf{Z}^k)
$$

$$
= \sum_{b=\max(0,-d)}^{\min(r^{t-1},e^t-d)} \left(\binom{r^{t-1}}{b} \sum_{j=1} \bar{\kappa}_{b,j} q \right.
$$

$$
\left. \times \left(\mathbf{P}^t_{-\{r^{t-1}+1-b,\ldots,r^t\}} \mid \mathbf{P}^{t-1}_{-U_b^{r^{t-1}}(j)}, \mathbf{Z}^t \right) \right)
$$

$$
\times \left(\binom{e^t}{d+b} \sum_{j=1} \bar{\nu}_{d+b,j} \prod_{l=1}^{d+b} U_{V_{\mu_{d+b}^{A^t}(l,j)}} \left(\boldsymbol{\rho}^t_{r^{t-1}+l-b} \right) \right) \quad (54)
$$

where

$$
\bar{\nu}_{a,j} = \nu_{a,j} \Big/ \sum_{i=1}^{\binom{e^t}{a}} \nu_{a,i}
$$

$$
\bar{\kappa}_{b,j} = \kappa_{b,j} \Big/ \sum_{i=1}^{\binom{r^{t-1}}{b}} \kappa_{b,i}. \quad (55)
$$

The importance density (54) has the same structure as the transition density (45). However, mixture components in the importance density, which correspond to the different ways of adding and removing targets, are weighted according to the existence probabilities and the states of persistent targets are drawn from the JOID rather than the prior target dynamics.

The weight update implied by this choice of importance density can be found by introducing the auxiliary variables b, j_1 and j_2 which are indices on the number of targets to be removed, the combination of targets to be removed and the combination of targets to be added, respectively. Using (25), (42), and (46),

the transition density for the position elements of the extended multitarget state with these auxiliary variables is

$$
p(\mathbf{P}^t, r^t = r^{t-1} + d, b, j_1, j_2 \mid \mathbf{P}^{t-1}, r^{t-1})
$$

$$
= \rho_{b,j_1} p \left(\mathbf{P}^t_{-\{r^{t-1}+1-b,\ldots,r^t\}} \Big| \mathbf{P}^{t-1}_{-U_b^{r^{t-1}}(j)} \right)
$$

$$
\times \lambda_{d+b,j_2} \prod_{l=1}^{d+b} U_{V_{\mu_{d+b}^{A^t}(l,j_2)}} \left(\boldsymbol{\rho}^t_{r^{t-1}+l} \right) p_{v_o} \left(\mathbf{v}^t_{r^{t-1}+l} \right). \quad (56)
$$

A similar equation can be obtained for the importance density:

$$
q(\mathbf{P}^t, r^t = r^{t-1} + d, b, j_1, j_2 \mid \mathbf{P}^{t-1}, r^{t-1}, \mathbf{Z}^t)
$$

$$
= \kappa_{b,j_1} q \left(\mathbf{P}^t_{-U_b^{r^{t-1}}(j_1) \cup \{r^{t-1}+1,\ldots,r^t\}} \Big| \mathbf{P}^{t-1}_{-U_b^{r^{t-1}}(j_1)}, \mathbf{Z}^t \right)
$$

$$
\times \nu_{d+b,j_2} \prod_{l=1}^{d+b} U_{V_{\mu_{d+b}^{A^t}(l,j_2)}} \left(\boldsymbol{\rho}^t_{r^{t-1}+l} \right). \quad (57)
$$

Substituting (56) and (57) into (11) gives the weight update factor for the importance density that uses the JOID for persistent targets and the measurement-directed method of adding and removing new targets as

$$
\frac{p(\mathbf{z}^t \mid \mathbf{X}^t)}{\prod_{l=1}^{s} \pi \left(\mathbf{z}^t \mid \mathbf{c}^t_l \right)} \frac{\lambda_{d+b,j_2} \rho_{b,j_1}}{\nu_{d+b,j_2} \kappa_{b,j_1}} \quad (58)
$$

where $c^t_1, \ldots, c^t_s$ are the clusters of positions of the persistent targets $\mathbf{X}^t_{-\{r^{t-1}+1-b,\ldots,r^t\}}$.

A summary of the integrated tracking and detection algorithm is given in Table III. The main issue in the computer implementation is the maintenance of target correspondences between particles. The convention that all particles which initiate a target in a particular cell at a particular time are initiating the same target is adopted. If a new target is initiated in any particle, a new partition is added to all particles. Particles in which this target is added will place the proposed initial state in this partition while particles which do not initiate this target assign a placeholder to this partition. A given partition then corresponds to the same target in all particles so that state estimates, required for clustering, can be obtained. If a partition is empty in all particles it is removed. This procedure corresponds to adding a set of labels to the extended multitarget state. Given labels for each particle, denoted by l^t_k for the kth particle, and assuming a maximum of R^t targets at time t, the ith target state is estimated at time t as (59), shown at the bottom of the page, where $I_i \subseteq \{1, \ldots, n\}$ contains the indices of particles that contain the ith target. The

$$
\mathsf{E}(\mathbf{x}^t_i \mid \mathbf{Z}^t, i \text{ exists}) = \frac{\sum_{r^t=0}^{R^t} \sum_{j=1}^{\binom{R^t}{r^t}} \int \mathbf{x}^t_i \Pr \left(i \text{ exists} \Big| \mathbf{X}^t, l^t = U_{r^t}^{R^t}(j), r^t \right) p \left(\mathbf{X}^t, l^t = U_{r^t}^{R^t}(j), r^t \Big| \mathbf{Z}^t \right) d\mathbf{X}^t}{\sum_{r=0}^{R^t} \sum_{j=1}^{\binom{R^t}{r}} \int \Pr \left(i \text{ exists} \mid \mathbf{X}, l = U_r^{R^t}(j), r \right) p \left(\mathbf{X}, l = U_r^{R^t}(j), r \mid \mathbf{Z}^t \right) d\mathbf{X}}
$$

$$
\approx \frac{\sum_{k \in I_i} w^t_k \mathbf{x}^t_{i,k}}{\sum_{s \in I_i} w^t_s}. \quad (59)
$$

- Separate the targets into clusters using (12).

- Update the existence grid using (48) and (49).

- For each particle:

    - Remove targets from the particle according to the probabilities of (52).

    - For each target cluster:

        * Compute the first-stage weights for the given cluster and particle using (34).

- For each particle:

    - For each target cluster:

        * Select the particle to be used for the given cluster and particle using the first-stage weights.

        * Draw from the JOID conditional upon the selected particle using (28).

        * For each target of the given cluster and particle, posterior density of the velocity elements.

    - Add targets to the given particle according to the probabilities of (51).

    - Compute the second-stage weights using (58).

- Perform resampling to obtain an evenly weighted particle set.

procedure is applied with R^t equal to the total number of distinct targets proposed across all particles.

C. Simulation Results

The detection and tracking algorithm is tested using the scenario described in Section IV-D and illustrated in Fig. 3. Measurements are generated using the thresholded model described in Section II. No prior knowledge of the number of targets or their locations is assumed. Targets are observed for 250 time-steps of 1 s.

Target velocity is initialized as follows. When a target is added to a particle, draw two integers η_1, η_2 such that $\mathsf{Pr}(\eta_i = j) = 1/2$ for $j \in \{-1, 1\}$ and then select the initial distribution of the target velocity to be $N(10(\eta_1, \eta_2)', 9\mathbf{I}_2)$. The arrival and removal probabilities are assumed to be spatially and temporally constant and are set to $\alpha = 0.02$ and $\beta = 0.005$, respectively. The single target detection probability is set to $P_D = 0.5$ so that $P_{\mathrm{FA}} = 0.5^{1+\lambda}$, where λ is the SNR.

The performance of the algorithm is measured by the mean number of true and false targets in track at each time-step. Specifically, let R^t denote the total number of distinct targets across all particles at time t and

$$\varepsilon_i^t = \sum_{k \in I_i} w_k^t, i = 1, \ldots, R^t \qquad (60)$$

where $I_i \subseteq \{1, \ldots, n\}$ is the collection of particle indices in which the ith target exists. The quantity ε_i^t can be interpreted as the probability that the ith partition belongs to an actual target at time t. In the following analysis, the ith partition is said to belong to a target if $\varepsilon_i^t \geq \Xi$ where $\Xi \in [0, 1]$ is a user-defined threshold. Let $E^t = \{i \in \{1, \ldots, R^t\} : \varepsilon_i^t \geq \Xi\}$ denote the set of presumed targets. For $i \in E^t$, state estimates are computed as shown in (59). These estimates are matched up with the ground

truth to determine which of the presumed targets are true targets and which are false targets. A presumed target is said to be a true target if its position estimate lies within 150 m of the actual target position. At most one presumed target can be matched to each true target.

Fig. 4 shows the average number of true and false targets plotted against time for sample sizes from 25 particles to 500 particles with an SNR of 10 dB. The filter is quick to establish tracks on all targets for all sample sizes considered here. In particular, less than 15 time-steps are sufficient to achieve true target averages of close to ten. Variations in sample size mainly affect false track discrimination. For a sample size of 25 particles the average number of false targets considered across the surveillance period is never less than one and reaches as high as 3.5. Performance improves considerably as the sample size increases so that the average number of false tracks is less than 0.2 for most of the surveillance period when 500 particles are used. The slight dip in performance between times 150 and 200 corresponds to a period of time where two targets occupy the same detection cell. This drop in performance may be attributed to the subtlety with which the thresholded measurement model indicates the presence of multiple targets in a single cell. This information will be reflected only in that detections will be registered in a cell containing multiple targets with a higher probability than if only one target occupied the cell. It appears that larger sample sizes are required to resolve the difference between these two situations. The use of a nonthresholded measurement model or the incorporation of the sensor management techniques of [31] may enable the difficulty to be resolved with a smaller sample size.

Fig. 5 shows the average number of true and false targets plotted against time for SNRs from 4 to 10 dB with a sample size of 200 particles. It can be seen that variations in SNR mainly
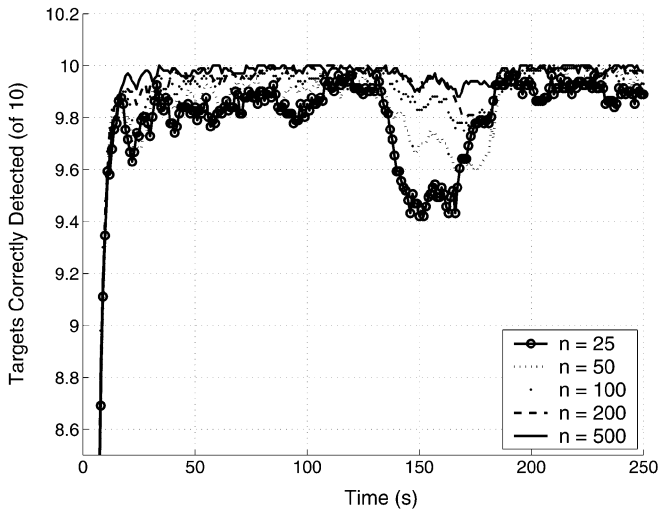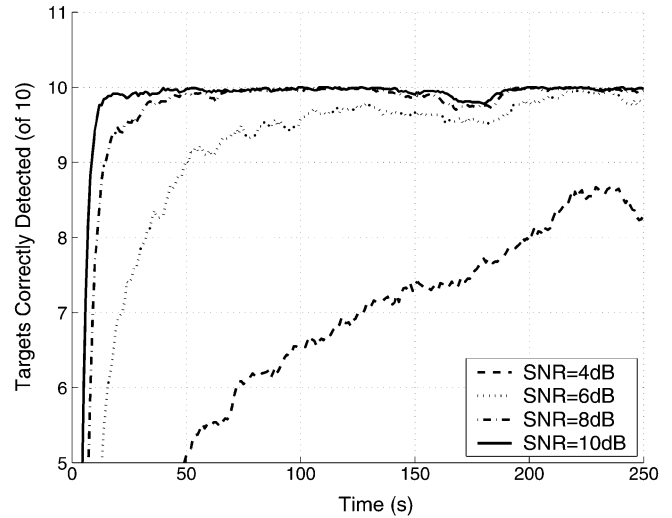
(a)
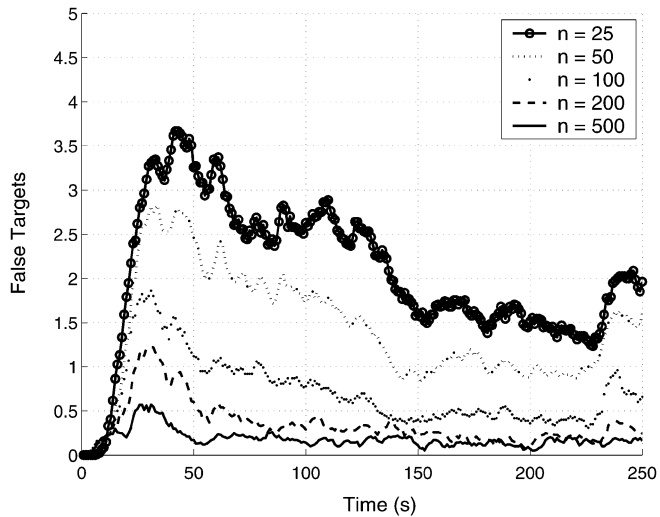
(b)

Fig. 4. Number of true and false targets detected by the proposed algorithm plotted against time for sample sizes from 25 particles to 500 particles. The arrival and removal probabilities were assumed spatially and temporally invariant and set at $\alpha = 0.02$ and $\beta = 0.005$, respectively. The SNR is 10 dB.
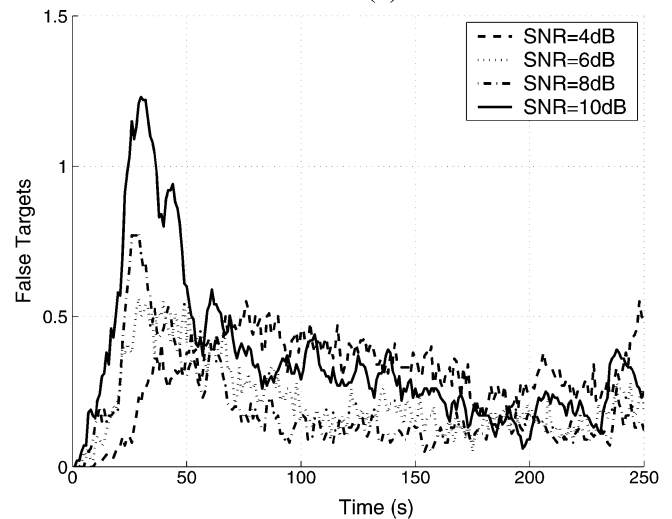

(a)

(b)

Fig. 5. Number of true and false targets detected by the proposed algorithm plotted against time for SNRs from 4 to 10 dB. The arrival and removal probabilities were assumed spatially and temporally invariant and set at $\alpha = 0.02$ and $\beta = 0.005$, respectively. The sample size is 200 particles.

effect the number of true targets detected. The false track discrimination is remarkably constant across the range of SNRs considered here. Excellent performance is achieved for SNRs greater than 8 dB with only a small drop in performance evident for a SNR of 6 dB. Even so, the results achieved for an SNR of 4 dB are quite good when it is considered that a single target detection probability of 0.5 results in a false alarm probability of 0.088 at this SNR. Significant improvements can be expected with larger sample sizes, particularly for small SNRs.

VI. CONCLUSION

A procedure based on approximation of the joint multitarget probability density was proposed for simultaneously detecting and tracking multiple targets. Approximation of the joint multitarget probability density is achieved by an efficient particle filtering scheme. The key elements of the filter design are clustering of targets based on position, joint sampling for targets

within clusters and measurement-directed placement of new targets and removal of existing targets. Simulation results attest to the performance of the proposed algorithm in a variety of conditions. The performance analyses were performed using a measurement model in which continuous-valued intensities generated in each cell of the surveillance region are thresholded to produce either a one, indicating the presence of a target/s, or a zero. The thresholded measurement model presents a more challenging tracking environment than the nonthresholded measurement model since a loss of information is incurred when thresholding is performed. Preliminary results indicate that significantly improved results are obtained using nonthresholded measurements.

Although we believe that the general formulation used here is applicable to a range of target dynamic and measurement models, some modifications specific to the particular models

under consideration would be required. Adaptation of the procedure to models of practical concern other than those used here would be of interest.

The arrival and removal probabilities required for the transition model of the extended multitarget state will, in many cases, have only a partial physical interpretation in the sense that the use of any available physical knowledge will not necessarily result in the best possible performance. Rather, the physical rate of target arrival acts as a nominal level about which the arrival probability parameter can be varied in order to improve performance. Similarly for the removal probability parameter. The natural question is then how to select these tuning parameters. The authors have noted that, within the range of values which may be deemed suitable, significant variations in performance arise. The possibility of selecting these parameters in a manner that is in some way optimal is well worth exploring.

The computational expense of the algorithm is essentially determined by the number of targets in the largest cluster. This suggests that the computational tractability of the algorithm should scale well in the number of targets, provided that large numbers of targets are not clustered together. If large numbers of targets are in close proximity computational expense can be reduced by either reducing the clustering threshold or increasing the neighborhood threshold. The deficiencies of the former alternative have been well documented in this paper. Although the latter alternative has not been investigated in depth by the authors, preliminary results suggest that the neighborhood threshold can be made reasonably large without affecting performance. Further investigation of the effect of varying the neighborhood threshold as a means of reducing computational expense is warranted.

The proposed algorithm allows only one target to be initiated in a measurement cell. This imposition is motivated by the subtle manner in which the sensor conveys target number information in each cell. A topic for future work will be to extract this information from the sensor returns and permit the initiation of multiple targets in a single measurement cell.

REFERENCES

[1] S. Blackman, *Multiple-Target Tracking with Radar Applications*. Norwood, MA: Artech House, 1986.

[2] Y. Bar-Shalom and W. D. Blair, *Multitarget-Multisensor Tracking: Applications and Advances*. Norwood, MA: Artech House, 2000.

[3] B. Ristic, M. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Norwood, MA: Artech House, 2004.

[4] Y. Barniv, "Dynamic programming solution for detecting dim moving targets," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 21, no. 1, pp. 144–156, 1985.

[5] S. Tonissen and Y. Bar-Shalom, "Maximum likelihood track-before-detect with fluctuating target amplitude," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 34, no. 3, pp. 796–809, 1998.

[6] L. D. Stone, C. A. Barlow, and T. Corwin, *Bayesian Multiple Target Tracking*. Norwood, MA: Artech House, 1999.

[7] M. I. Miller, A. Srivastava, and U. Grenander, "Conditional mean estimation via jump-diffusion processes in multiple target tracking/recognition," *IEEE Trans. Signal Process.*, vol. 43, no. 11, pp. 2678–2690, Nov. 1995.

[8] K. Kastella, "Event averaged maximum likelihood estimation and mean-field theory in multitarget tracking," *IEEE Trans. Autom. Control*, vol. 50, no. 6, pp. 1070–1073, 1995.

[9] R. Mahler, "'Statistics 101' for multisensor, multitarget fusion," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 19, no. 1, pp. 53–64, 2004.

[10] K. Kastella, "A maximum likelihood estimator for report-to-track association," in *Proc. SPIE*, 1993, vol. 1954, pp. 386–393.

[11] S. Musick, K. Kastella, and R. Mahler, "A practical implementation of joint multitarget probabilities," in *Proc. SPIE*, 1998, vol. 3374, pp. 26–37.

[12] E. W. Kamen, "Multiple target tracking based on symmetric measurement functions," *IEEE Trans. Autom. Control*, vol. 37, no. 3, pp. 371–374, 1992.

[13] S. Mori, C.-Y. Chong, E. Tse, and R. Wishner, "Tracking and classifying multiple targets without *a priori* identification," *IEEE Trans. Autom. Control*, vol. 31, no. 5, pp. 401–409, 1986.

[14] H. M. Shertukde and Y. Bar-Shalom, "Tracking of crossing targets with imaging sensors," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 27, no. 4, pp. 582–592, 1991.

[15] K. Kastella, "Discrimination gain for sensor management in multitarget detection and tracking," in *Proc. IMACS Conf. Computational Engineering Systems Applications*, Lille, France, 1996, pp. 167–172.

[16] C. M. Kreucher, K. Kastella, and A. O. Hero, "Tracking multiple targets using a particle filter representation of the joint mulitarget proabbility density," in *Proc. SPIE Conf. Signal Data Processing of Small Targets*, 2003, pp. 258–269.

[17] C. Kreucher, K. Kastella, and A. Hero, "Multitarget tracking using a particle filter representation of the joint multitarget probability density," *IEEE Trans. Aerosp. Electron. Syst.*, accepted for publication.

[18] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[19] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*. New York: Springer-Verlag, 2001.

[20] R. Bucy and K. Senne, "Realization of optimum discrete-time nonlinear estimators," in *Proc. Symp. Nonlinear Estimation Theory Its Applications*, San Diego, CA, 1971, pp. 6–17.

[21] Y. Boers and J. Driessen, "Multitarget particle filter track before detect application," *Proc. Inst. Elect. Eng.—Radar, Sonar, Navigation*, vol. 151, no. 6, pp. 351–357, 2004.

[22] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Stat. Comput.*, vol. 10, pp. 197–208, 2000.

[23] M. Morelande, C. Kreucher, and K. Kastella, "Multiple target tracking with a pixelized sensor," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, Philadelphia, PA, 2005, vol. V, pp. 585–588.

[24] M. Orton and W. Fitzgerald, "A Bayesian approach to tracking multiple targets using sensor arrays and particle filters," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 216–223, Feb., 2002.

[25] G. Casella and C. Robert, "Rao–Blackwellisation of sampling schemes," *Biometrika*, vol. 83, no. 1, pp. 81–94, 1996.

[26] C. Kreucher, A. Hero, and K. Kastella, "Multiple model particle filtering for multitarget tracking," presented at the Conf. Adaptive Sensor Array Processing, Lexington, MA, Mar. 16–18, 2004.

[27] M. Morelande and S. Challa, "Manoeuvring target tracking in clutter using particle filters," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 1, pp. 252–270, 2005.

[28] G. Stimson, *Introduction to Airborne Radar*, 2nd ed. Mendham, NJ: SciTech Publishing, 1998.

[29] W. Blair and M. Brandt-Pearce, "Unresolved Rayleigh target detection using monopulse measurements," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 34, no. 2, pp. 543–552, 1998.

[30] C. Gowda and R. Viswanatha, "Performance of distributed CFAR test under various clutter amplitudes," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 35, no. 4, pp. 1410–1419, 1999.

[31] C. Kreucher, K. Kastella, and A. Hero, "Sensor management using an active sensing approach," *Signal Process.*, vol. 85, no. 3, pp. 607–624, 2005.

[32] D. Crisan and A. Doucet, "A survey of convergence results on particle filtering methods for practitioners," *IEEE Trans. Signal Process.*, vol. 50, no. 3, pp. 736–746, Mar. 2002.

[33] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *Proc. Inst. Elect. Eng.—F*, vol. 140, no. 2, pp. 107–113, 1993.

[34] A. Doucet, N. Gordon, and V. Krishnamurthy, "Particle filters for state estimation of jump Markov linear systems," *IEEE Trans. Signal Process.*, vol. 49, no. 3, pp. 613–624, Mar. 2001.

[35] Y. Bar-Shalom and X.-R. Li, *Estimation and Tracking: Principles, Techniques and Software*. Norwood, MA: Artech House, 1993.

[36] M. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *J. Amer. Stat. Assoc.*, vol. 94, pp. 590–599, 1999.

[37] M.-S. Oh, "Monte Carlo integration via importance sampling: Dimensionality effect and an adaptive algorithm," *Contemp. Math.*, vol. 115, pp. 165–187, 1991.

[38] M. Morelande, C. Kreucher, and K. Kastella, "A study of factors affecting multiple target tracking with a pixelised sensor," presented at the SPIE Conf. Signal Data Processing of Small Targets, San Diego, CA, Aug. 2–4, 2005.
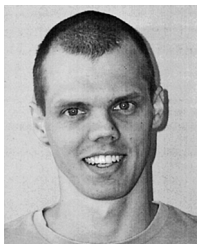
Mark R. Morelande received the B.Eng. degree in aerospace avionics from Queensland University of Technology, Brisbane, Australia, in 1997 and the Ph.D. degree in electrical engineering from Curtin University of Technology, Perth, Australia, in 2001.

From November 2000 to January 2002, he was a Postdoctoral Fellow at the Centre for Eye Research, Queensland University of Technology. Since 2002, he has been a Research Fellow at the Department of Electrical and Electronic Engineering, University of Melbourne, Australia. His research interests include nonstationary signal analysis and target tracking with particular emphasis on multiple target tracking and sequential Monte Carlo methods.

Christopher M. Kreucher received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of Michigan, Ann Arbor, in 1996, 1997, and 2005, respectively.

Since 1998, he has been a Principal Scientist at General Dynamics Advanced Information Systems' Michigan Research and Development Facility (formerly ERIM), Ypsilanti, MI. His current research interests include nonlinear filtering (specifically particle filtering), Bayesian methods of multitarget tracking, self localization, information theoretic sensor management, and distributed swarm management.
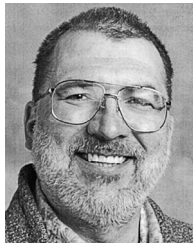
Keith Kastella received the B.A. degree in physics from Reed College, Portland, OR, in 1980 and the M.S. and Ph.D. degrees from the State University of New York at Stony Brook (SUNY-SB) in 1985 and 1988, respectively.

He is a Chief Scientist at General Dynamics Advanced Information Systems, Ypsilanti, MI. His current research interests are in data fusion, signal processing, and novel applications of quantum entanglement in sensing and signal processing.