Multi-target Tracking for Measurement Models with Additive Contributions

Frederic Thouin, Santosh Nannuru, Mark Coates
Electrical and Computer Engineering Department
McGill University
Montreal, Canada
Email: frederic.thouin@mail.mcgill.ca, santosh.nannuru@mail.mcgill.ca, mark.coates@mcgill.ca

***Abstract*—Moment-based filters, such as the Probability Hypothesis Density (PHD) filter, are an attractive solution to multi-target tracking. However, an underlying assumption for the PHD filter is that each measurement is either caused by a single target or clutter. In this paper, we design a novel moment-based multi-target filter, the Additive Likelihood Moment (ALM) filter, where the measurements are affected by all targets. We focus on the cases where the likelihood can be expressed as a function of the sum of the individual target contributions. As an example, we consider radio tomographic tracking where the attenuation of the signal between a pair of sensors is the sum of attenuations caused by all targets. Our multi-target tracking algorithm is based on a particle approximation of our moment-based filter. Our simulations show that our algorithm has a lower estimation error than MCMC particle methods while achieving 85% savings in terms of computational time.**

Keywords: Tracking, filtering, clustering.

I. Introduction

Tracking multiple targets is very challenging due to the high dimensionality of the state-space. A generalization of the recursive Bayes filter [1] is the optimal approach, but it is generally not practical due to its computational complexity. An attractive alternative is moment-based filters, such as the Probability Hypothesis Density (PHD) filter [2]. The PHD filter tracks the first-order moment of the multi-target posterior rather than the posterior itself. This reduces the dimensionality of the problem to a single target state space, which is easier to track than the joint probability distribution. A limitation of this filter is the assumption that (i) each target causes either one or no measurement and (ii) each measurement is either caused by a single target or clutter.

We are interested in developing a moment-based multi-target filter for applications where all targets have an influence on measurements (i.e., measurements are a function of all targets) such as direction of arrival for linear antenna arrays [3], multi-user detection for wireless communication networks [4] or radio frequency (RF) tomographic tracking [5]. We will use the last-mentioned as an example throughout this paper. RF tomography is a passive and device-free method that can be used to estimate the position of moving targets in an observed area surrounded by wireless sensors. The Received Signal Strength (RSS) of an RF signal is attenuated by obstructions in the monitored area. Therefore, a greater attenuation between a pair of sensors can indicate the presence of a moving object. In this environment, (i) each target can contribute to any number of measurements, (ii) each measurement is potentially affected by multiple targets and (iii) measurements are not independent.

In this paper, we design a novel moment-based multi-target filter, called the Additive Likelihood Moment (ALM) filter, for scenarios where the likelihood model can be expressed in the form $L(x) = h(\sum_i g(x_i))$. Here h and g are non-linear functions, satisfying some technical conditions detailed later. The key aspect in this formulation is that the targets affect the measurement in an additive fashion, so that the likelihood value can be determined from the *sum* of the individual effects. We focus on the special case where h is a Gaussian function because it allows us to derive a computationally efficient filter. The rest of this paper is organized as follows. In Sect. II, we very briefly review existing methods used for multi-target tracking. In Sect. III, we formally define the multi-target tracking problem in the context of RF tomography. In Sect. IV, we derive equations for our novel ALM filter. In Sect. V, we describe our implementation of an existing MCMC algorithm for multi-target tracking. In Sect. VI, we present the results of our Matlab simulations and use the MCMC algorithm as a benchmark for the ALM filter. In Sect. VII, we summarize our work and suggest future directions of exploration.

II. Related Work

There is a large amount of literature on multi target tracking. In this section, we cite the most important ones which are relevant to our work. For a more complete survey, see [6]. Sequential Monte Carlo (SMC) methods have proven effective for non-linear target tracking but their performance is known to deteriorate when the dimension of the state vector is large [7]. Thus, as an alternative to the SMC methods, Markov Chain Monte Carlo (MCMC) based particle methods have been suggested for tracking problems of high dimensionality [8], [9].Khan et al. construct a Markov chain to sample from the posterior state distribution at every time step k, but their method is computationally expensive [8]. Pang et al. use a more efficient approach and sample from the combined state at time k and $k-1$ [9].

Another approach to reduce the complexity of the problem is moment-based filters, such as the PHD filter, that propagate the posterior expectation rather than the joint multi-target posterior distribution [2]. Although most successful

implementations of the PHD filter for multi-sensor multi-target tracking [10]–[12] are based on the assumption that measurements are caused by a single target or clutter, there are a few examples where this is not the case [3], [13]. Balakumar et al. perform direction of arrival tracking for linear antenna arrays that have an observation model with additive contributions [3]. Their approach is (i) limited to narrowband, far-field sources of the linear sensor array and (ii) require the SMC-PHD filter input to be first processed using discrete Fourier transform techniques. To the best of our knowledge, Mahler's work on superpositional sensors is the most related to ours [13]. He derives corrector equations for the cardinalized PHD filter based on a multitarget measurement model with the same form as the likelihood model we require. However, the equations include multi-dimensional integrals, which are computationally intractable. The filter we present in this paper applies to the superpositional sensors model and is tractable form (single-target state space).

III. Problem Statement

We are interested in a general multi-target tracking problem where the multi-target state is represented as a finite subset X_k. If targets are in the single-target space E_s then,

$$X_k = \{x_{k,1}, \ldots, x_{k,N_t(k)}\} \in \mathcal{F}(E_s) \tag{1}$$

is the multi-target state, where $\mathcal{F}(E)$ denotes the collection of all finite subsets of the space E and $N_t(k)$ is the number of targets at time k. Similarly, Z_k is a finite set that represents multi-target measurement at time k. If there are M observations $z_{k,1}, \ldots, z_{k,M}$ in the single-target observation space E_o are received at time k, then

$$Z_k = \{z_{k,1}, \ldots, z_{k,M}\} \in \mathcal{F}(E_o) \tag{2}$$

is the multi-target measurement. We assume that the targets move according to a Markovian dynamic model in the form $\mathbf{x}_{k+1,i} = f_{k+1|k}(\mathbf{x}_{k,i}, \mathbf{u}_k)$ where $\mathbf{u}_k$ is noise.

Our formulation differs from most multi-sensor multi-target problem statements because we consider the case where targets affect measurements in an additive manner. Also, we assume that we receive a constant number of measurements, M, at each time step. More formally, the likelihood model can be expressed in the form

$$L(X_k) = h\left(\sum_{x_i \in X_k} g(x_i)\right) \tag{3}$$

where g and h are non-linear functions. Also, g satisfies:

$$\int_S g(\mathbf{x})^2 \lambda(\mathbf{x}) d\mathbf{x} < \infty \tag{4}$$

where $\lambda(\mathbf{x})$ is the intensity of the Poisson process used to approximate the predictive posterior (see App. A).

A. Example: Radio-frequency tomography

Throughout this paper, we use radio-frequency (RF) tomography as an example where the likelihood model fits this form. The scenario we consider is tracking a fixed number of targets, $N_t(k) = N_t$, $\forall k$, moving according to a known model inside a $L \times L$ observation space surrounded by RF sensors located on the edge. One such scenario is depicted in Figure 1.

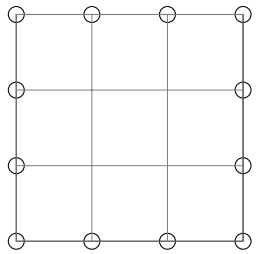


Figure 1. Monitoring area with size $L = 3$, $N = 12$ sensors (represented by circles on the area boundaries) and $M = 66$ links.

At every time instant k, the sensors communicate with each other and record the Received Signal Strength (RSS) values. The N sensors have a total of $M = N(N-1)/2$ communication links generating M measurements in every time step. During a period with an empty surveillance region, the sensor system learns *background* RSS values for each link. The objective of RF tomography is to use the measured deviations from these background RSS values to track moving targets inside the observation field. RF tomography avoids some of the more challenging calibration issues associated with other localization techniques based on RSS measurements because it is based on these deviations rather than the raw RSS measurements. Due to space limitations, we refer the readers to [14] for more details about the sensors and RF tomography.

In our case, the targets move within the boundaries of the monitoring area according to a linear Gaussian dynamics [11]:

$$x_{k+1,i} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x_{k,i} + \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ T & 0 \\ 0 & T \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix}$$

where T is the sampling period and u_x, u_y are zero-mean Gaussian white noise with respective variance σ_{u_x} and σ_{u_y}. In this model, the state of each object i at time k, $x_{k,i}$, is represented by a four-dimensional vector: position on the x-axis and y-axis, velocity on the x-axis and y-axis.

In [14], a single target measurement model has been proposed for RF tomography based on experimental analysis. The mean RSS attenuation on the link j due to a target at position x at time k is modelled as:

$$g_k^j(x) = \phi \exp\left(-\frac{\lambda_j^2(x)}{2\sigma_\lambda^2}\right) \tag{5}$$

where $\lambda_j(x)$ is the distance between a target located at x and link j; ϕ and σ_λ are known and fixed parameter based on physical properties of the sensors that have been learned

empirically. An intuitive justification for the model is that a target located far from link j has a large associated $\lambda_j(x)$ and hence causes minimal additional attenuation $g_k^j(x)$; when the target is close to the link it has a much greater impact.

We extend this model to multiple targets case by using the fact that the signal attenuation due to multiple targets is the sum of attenuation due to each of the individual target. Thus according to this model if $g_k^j(x_i)$ is the attenuation on link j due to the target with state x_i, then the total attenuation on link j due to all the targets combined would be given by

$$g_k^j(\mathbf{x}) = \sum_{i=1}^{N_t(k)} g_k^j(x_i) \tag{6}$$

The observed noisy measurement at time step k is given by

$$Z_k = g_k(\mathbf{x}) + \sigma_z S_k \tag{7}$$

where $g_k = [g_k^1\ g_k^2 \cdots g_k^M]$ and S_k is the noise assumed to be distributed according to $\mathcal{N}(0, I_{M\times M})$. This form of measurement allows the likelihood to be expressed as $h\left(\sum_i g(x_i)\right)$.

IV. ALM Filter Algorithm

A. ALM Filter

The notation $D_{k+1|k+1}(\mathbf{x}) = D_{k+1|k+1}(\mathbf{x}|Z^{(k+1)})$ denotes the posterior PHD at state value $\mathbf{x}$, calculated at time $k+1$ using all measurements up to time $k+1$. The notation $D_{k+1|k}(\mathbf{x}) = D_{k+1|k}(\mathbf{x}|Z^{(k)})$ denotes a *predictive* PHD, i.e. the PHD calculated at time $k+1$ using the data up to time k. The likelihood function at time k is denoted $f_k(Z|X)$. Set integrals are indicated using the notation $\int \cdot\, \delta W$; integrals over the state space are denoted $\int \cdot\, d\mathbf{x}$.

Since our assumption about the likelihood model and the measurements does not affect the time prediction step of the filter, we can apply Mahler's general law of motion for PHDs to compute the predictive PHD [2]:

$$\begin{aligned} D_{k+1|k}(\mathbf{x}) &= b_{k+1|k}(\mathbf{x}) \\ &+ \int \left[p_s(\mathbf{w}) f_{k+1|k}(\mathbf{x}|\mathbf{w}) + b_{k+1|k}(\mathbf{x}|\mathbf{w})\right] D_{k|k}(\mathbf{w}) d\mathbf{w} \end{aligned} \tag{8}$$

where $b_{k+1|k}(Y)$ is the likelihood that new targets with state-set Y enter the monitoring area at time-step $k+1$, $p_s(\mathbf{x})$ is the probability that a target with state $\mathbf{x}$ at time k will survive in time-step $k+1$ and $b_{k+1|k}(Y|\mathbf{x})$ is the likelihood that a group of new targets with state-set Y will be spawned at time-step $k+1$ by a single target that had state $\mathbf{x}$ at time k.

On the other hand, we cannot apply Mahler's single-sensor Bayes update formula for PHD directly. We start from the general definition of a multi-target moment density:

$$D_{k+1|k+1}(\mathbf{x}) = \int f_{k+1|k+1}(\{\mathbf{x}\} \cup W|Z^{(k+1)}) \delta W \tag{9}$$

The key ingredient in deriving the update expression is the form of our likelihood:

$$f_k(Z_k|W) = h_Z\left(\sum_{\mathbf{x}_i \in W} g_k(\mathbf{x}_i)\right) \tag{10}$$

This formulation allows us to use Campbell's theorem [15] to replace the set integral over the entire multi-target space with an integral over the single-target space. Then, we define $s = \sum_{\mathbf{x}_i \in W} g_k(\mathbf{x}_i)$ and adopt a Gaussian approximation[1] to $P(s)$, which allows us to write this general PHD update equation:

$$D_{k+1|k+1}(\mathbf{x}) = \frac{D_{k+1|k}(\mathbf{x})}{F_{k+1}(\mathbf{x})} \tag{11}$$

where

$$F_{k+1}(\mathbf{x}) \approx \frac{\int_0^\infty h_Z(s)\mathcal{N}_s(\mu_s, \Sigma) ds}{\int_0^\infty h_{Z'}(s)\mathcal{N}_s(\mu_s, \Sigma) ds} \tag{12}$$

$$\mathcal{N}_x(\mu, \Sigma) = \exp\left[-\frac{1}{2}(x-\mu)\Sigma^{-1}(x-\mu)\right] \tag{13}$$

In general, computing $F_{k+1}(\mathbf{x})$ is still not trivial. However, if h_Z is Gaussian (as in the RF tomography example), then the update expression reduces to this ratio of Gaussians:

$$F_{k+1}(\mathbf{x}) \approx \frac{\mathcal{N}_{Z_k}(\mu_s, \Sigma_s + \Sigma)}{\mathcal{N}_{Z_k}(g(\mathbf{x}) + \mu_s, \Sigma_s + \Sigma)} \tag{14}$$

For a complete proof, see the Appendix.

B. Particle approximation

Even after this approximation, the mean and variance remain unknown. We construct a particle approximation for the density function $D_{k|k}$ associated with the multi-target posterior $p_{k|k}$, which allows to form a particle approximation for μ_s and Σ. In this context, the interpretation is that the weight $w^{(i)}$ associated with each particle $x^{(i)}$ represents the probability that a target is present at the position of that particle. Instead of a continuous density function, we have a finite, chosen set of N_p positions inside the state space. The particle approximation for the posterior PHD is:

$$\widehat{D}_{k|k}(\mathbf{x}_k) = \sum_{i=1}^{N_p(k)} w_k^{(i)} \delta_{x_k^{(i)}}(x_k) \tag{15}$$

Now, we derive the particle approximations for the time prediction equation in (8) and the update equation in (11). We use a particle approximation for the predictive PHD which is identical to the one derived by Vo et al. (see [11] for more details). Let p_{k+1} and q_{k+1} be proposal densities, γ_{k+1} the intensity function of the spontaneous birth and $\phi_{k+1|k}(x, w) = p_s(w) f_{k+1|k}(x|w) + b_{k+1|k}(x|w)$. Then we can write

$$\widehat{D}_{k+1|k}(\mathbf{x}_{k+1}) = \sum_{i=1}^{N_p(k)+J_{k+1}} w_{k+1|k}^{(i)} \delta_{x_{k+1}^{(i)}}(x_{k+1}) \tag{16}$$

where

$$x_{k+1}^{(i)} \sim \begin{cases} q_{k+1}(\cdot|x_k^{(i)}, Z_k),\ i = 1, \ldots, N_p(k) \\ p_{k+1}(\cdot|Z_k),\ i = N_p(k)+1, \ldots, N_p(k)+J_{k+1} \end{cases}$$

[1]The authors of [16], [17] successfully use a Gaussian approximation for the aggregate interference (generated by interferers modelled by a Poisson process) in a CDMA network.

$$w_{k+1|k}^{(i)} = \begin{cases} \frac{\phi_{k+1|k}(x_{k+1}^{(i)}, x_k^{(i)}) w_k^{(i)}}{q_k(x_{k+1}^{(i)}|x_k^{(i)}, Z_k)}, & i = 1, \ldots, N_p(k) \\ \frac{1}{J_{k+1}} \frac{\gamma_{k+1}(x_{k+1}^{(i)})}{p_{k+1}(x_{k+1}^{(i)}|Z_k)}, & i = N_p(k)+1, \ldots, \\ & N_p(k) + J_{k+1} \end{cases}$$

In the example we are considering in this paper, we are assuming a fixed number of targets, i.e. no births, spawning and disappearances (survival probability is one). In that case, the number of particles is fixed such that $N_p(k) = N_p$ and $J_{k+1} = 0$. We choose $q_{k+1} = f_{k+1|k}$ and, therefore,

$$x_{k+1}^{(i)} \sim f_{k+1|k}(\cdot|x_k^{(i)}) \quad i = 1, \ldots, N_p$$

Furthermore, we have $\phi_{k+1|k}(x, w) = f_{k+1|k}(x|w)$ and the weight prediction reduces to

$$w_{k+1|k}^{(i)} = w_k^{(i)} \quad i = 1, \ldots, N_p$$

The approximation for the updated posterior is:

$$\widehat{D}_{k+1|k+1}(\mathbf{x}_{k+1}) = \sum_{i=1}^{N_p} w_{k+1}^{(i)} \delta_{x_{k+1}^{(i)}}(x_{k+1}) \tag{17}$$

where

$$w_{k+1}^{(i)} = \frac{w_{k+1|k}^{(i)}}{F_{k+1}(x_{k+1}^{(i)})} \tag{18}$$

We use the following particle approximation for μ_s and Σ:

$$\mu_s \approx \sum_j w_{k+1|k}^{(j)} g(x_{k+1}^{(j)}) \tag{19}$$

$$\Sigma_{(a,b)} \approx \sum_l w^{(l)} g_a(x_{k+1}^{(l)}) g_b(x_{k+1}^{(l)}) \tag{20}$$

C. Algorithm

1 initialize particles;
2 **while** *tracking* **do**
3 **foreach** *particle* **do**
4 propagate particle;
5 update weight;
6 **end**
7 cluster particles with k-means;
8 resample (if necessary);
9 **end**

Algorithm 1: ALM algorithm

Our ALM filter algorithm is outlined in Alg. 1. First, we initialize our set of particles (line 1). The total number of particles N_p is chosen as a function of the number of targets N_t: $N_p = N_t \cdot N_{ppt}$. The particles are drawn from a distribution p_0. In the RF tomography case, the initial x and y positions of each particle are drawn randomly from a uniform distribution $\mathcal{U}(0, L)$ and the velocities from a zero-mean Gaussian with unit variance. At each iteration, we propagate the particles (line 4) according to the prediction equation in (16). Then, we update the weights (line 5) based on (18) using the new set of measurements. We estimate the targets' state (line 7) using k-means clustering [18]. If the effective number of particles $(\sum_{i=1}^{N_p} (w_k^{(i)})^2)^{-1}$ is less than a threshold, then we need to sample a new set of particles (line 8). Importance resampling is done by drawing N_p particles from the current particle set by using probabilities proportional to the weights $w_k^{(i)}$.

D. Computational complexity

The particle initialization (line 1) implies drawing samples from p_0 for each particle, which is $\mathcal{O}(N_p)$. At each iteration, we need to propagate the particles (line 4) and update their weights (line 5). The propagation is a simple multiplication and is $\mathcal{O}(N_p)$. The weight update involves estimating the covariance matrix (equation 20) which is $\mathcal{O}(N_p M^2)$, and computing the inverse of an $M \times M$ matrix which has a complexity $\mathcal{O}(M^3)$. The complexity of the k-means algorithm is $\mathcal{O}(N_t \times N_p \times I \times R)$ where I is the number of iterations (variable) and $R = 50$ is the number of times the clustering is repeated with different initial centroids. Finally, the complexity of importance resampling is $\mathcal{O}(N_p)$. The overall complexity for one iteration is:

$$\mathcal{O}(N_p + N_p M^2 + M^3 + 50 N_t N_p I + N_p) \approx \mathcal{O}((N_p + M) M^2)$$

V. MCMC Algorithm

In this section, we briefly review an MCMC based particle method for target tracking [7], [9]. At each time k the posterior distribution is approximated using a set of unweighted samples (particles) obtained from the MCMC procedure. The Metropolis Hastings algorithm is used to sample from the combined posterior distribution of X_k and X_{k-1} given by

$$p(X_k, X_{k-1}|Z^{(k)}) = \frac{p(Z_k|X_k) p(X_k|X_{k-1}) p(X_{k-1}|Z^{(k-1)})}{p(Z_k|Z^{(k-1)})} \tag{21}$$

Since we do not have the exact representation for $p(X_{k-1}|Z^{(k-1)})$, we use its particle representation obtained from the previous time step to approximate it. Assume that at time $(k-1)$ the unweighted particle representation of $p(X_{k-1}|Z^{(k-1)})$ is given by

$$p(X_{k-1}|Z^{(k-1)}) \approx \frac{1}{N_p} \sum_{j=1}^{N_p} \delta(X_{k-1} - X_{k-1}^{(j)}) \tag{22}$$

We use the two step algorithm outlined in [7] with the joint draw and refinement steps. The proposal distribution q_1 for the joint draw at the m^{th} iteration is chosen as

$$q_1(X_k, X_{k-1}|X_k^m, X_{k-1}^m) \propto \sum_{j=1}^{N_p} p(x_{k,i}|X_{k,\underline{i}}^m, X_{k-1}^{(j)}) \delta(X_{k-1}^{(l)} - X_{k-1}^{(j)}) \tag{23}$$

where l is chosen uniformly in $\{1, 2 \cdots N_p\}$ and i is chosen uniformly in $\{1, 2 \cdots N_t\}$. The expression $X_{k,\underline{i}}$ refers to the state vector of all the targets excluding target i. Thus in the joint draw step, multi-target state update is done only

for one randomly selected target per iteration. The proposal distribution q_3 for the refinement step is chosen as follows

$$q_3(X_k|X_k^m, X_{k-1}^m) \propto p(x_{k,i}|X_{k,\underline{i}}^m, X_{k-1}^m) \tag{24}$$

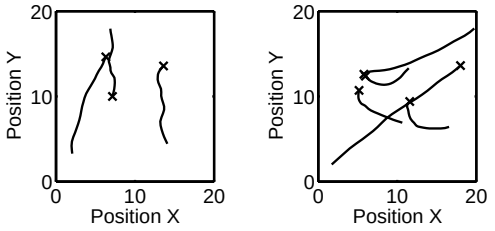
The same target i whose state is updated in the joint draw step is refined in this step.

At every time step, the Markov chain is initiated using the most likely particle chosen among all the particles from the previous time step. The algorithm described in [7] updates all the target states in both the steps which gives an order $O(N_t^2 M N_p)$ algorithm. In the current implementation, we randomly choose a target and perform the steps for only that target. This reduces the complexity of the algorithm to $O(N_t M N_p)$ and also gives better tracking results.

VI. Simulations

A. Settings

For the simulations, we arbitrarily choose an observation field of dimension $L = 20$. To simulate the target dynamics, we choose two scenarios with different number of targets ($N_t = 3$ and $N_t = 5$) displayed in Fig. 2. The objects move within the boundaries of the monitoring area according to a linear Gaussian dynamics (as described in Sect. III) where $T = 0.3$, $\sigma_{u_x} = 1$ and $\sigma_{u_y} = 0.5$. We track the targets for $K = 25$ time steps. For the RF tomography likelihood model described by (5), we set $\phi = 5$, $\sigma_\lambda = 0.05$, $\sigma_z = 1$. These values were chosen based on experimental studies described in [19]. We simulate for different numbers of sensors $N = [12, 28, 40]$ ($M = [66, 378, 780]$) and numbers of particles per target $N_{ppt} = [250, 500, 750, 1000]$. For MCMC simulations, we set $N_{burn} = 1000$ and $N_{thin} = 3$. For ALM simulations, we use $N_{eff} = N_p/7$. We generate 20 different sets of measurements for each configuration ($[N_t, N, N_{ppt}]$) and repeat 5 times (total of 100 runs per configuration).



(a) 3 targets (b) 5 targets

Figure 2. Target trajectories. The 'x' indicate the position at $t = 0$.

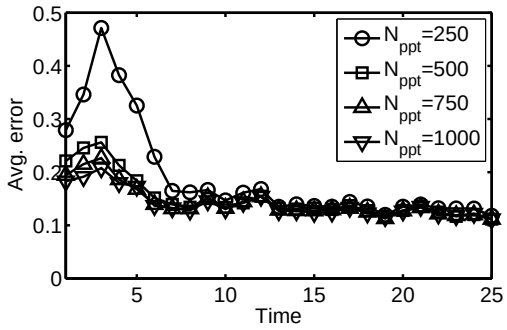
The error between the estimated positions of the objects and the ground truth is computed using the OSPA metric [20]. Since, we are currently assuming that we know the number of objects, the metric reduces to the p-th order OMAT metric:

$$d_p(X, Y) = \left(\frac{1}{n} \min_{\pi \in \Pi} \sum_{i=1}^{n} d(x_i, y_{\pi(i)})^p \right)^{1/p}$$

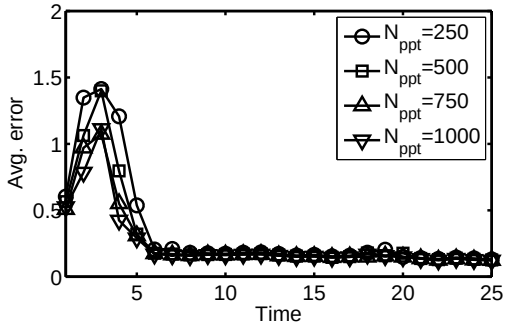
where Π is the set of permutations of $\{1, 2, \ldots, n\}$, $d(x, y)$ is the Euclidean distance between x and y, $X = \{x_1, \ldots, x_n\}$ and $Y = \{y_1, \ldots, y_n\}$ are arbitrary sets and p is a fixed parameter. For these simulations, we use $p = 2$. We compute the average error and observe the computational time for both the ALM and MCMC algorithms.

B. Results

In Fig. 3, we show the impact of the number of particles used per target on the average error for the ALM approach when $M = 378$ measurements are available. We observe that using $N_{ppt} = 250$ is slightly less accurate in the scenario with three targets. However, in the case of five targets, the algorithm appears to be less sensitive to the number of particles per target. It is possible that the algorithm accuracy depends on the total number of particles, not necessarily the number per target. Furthermore, the Gaussian approximation might be slightly less accurate for fewer targets.



(a) 3 targets

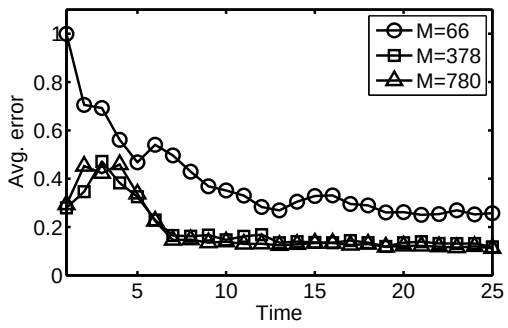


(b) 5 targets

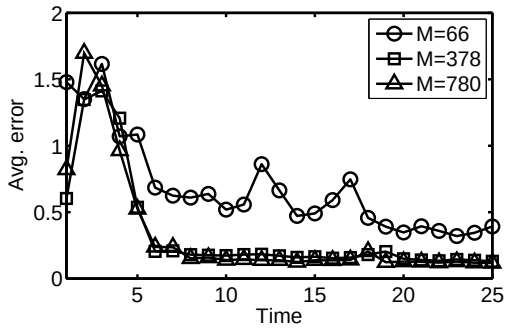
Figure 3. Average error of ALM (over 100 runs) as a function of time for $N_{ppt} = [250, 500, 750, 1000]$. The results are shown for $M = 378$ links.

In Fig. 4, we see that when the sensor density is too small, the number of measurements is insufficient to provide accurate estimates. In both scenarios, increasing the number of measurements from 378 to 780 provides no gain in accuracy and results in a longer computational time.

In Figs. 5 and 6, we compare average error of ALM to two versions of the MCMC algorithm. In Fig. 5, initial particles location are drawn randomly from a uniform and velocities from a zero-mean Gaussian with unit variance. With this initialization, the MCMC simulations give high average error. This is because of the high state dimension $4N_t$ that makes it

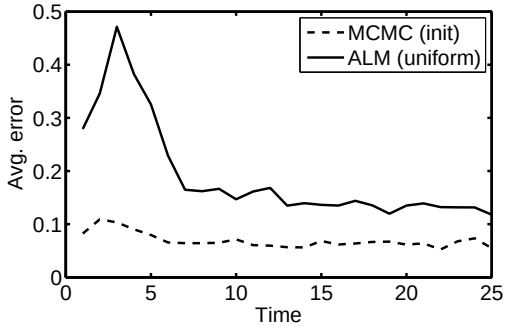


(a) 3 targets

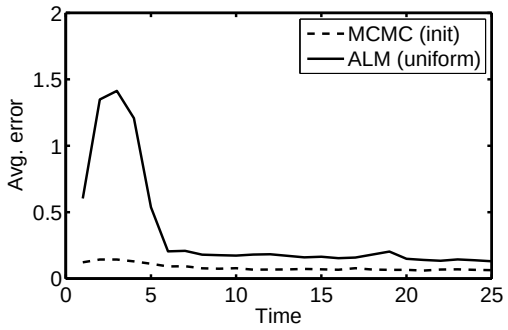


(b) 5 targets

Figure 4. Average error of ALM (over 100 runs) as a function of time when the number of links is varied. The results are shown for $N_{ppt} = 250$.



(a) 3 targets



(b) 5 targets

Figure 6. Average error (over 100 runs) of ALM and the MCMC algorithm with initial particles positions near actual target positions as a function of time. The results are shown for $N_{ppt} = 250$ and $M = 378$.

difficult to be sampled accurately. Thus particles different from the actual target state are used to represent the posteriori and as the targets move the error increases with time. We noticed the same behaviour for the scenario with $N_t = 5$ targets.

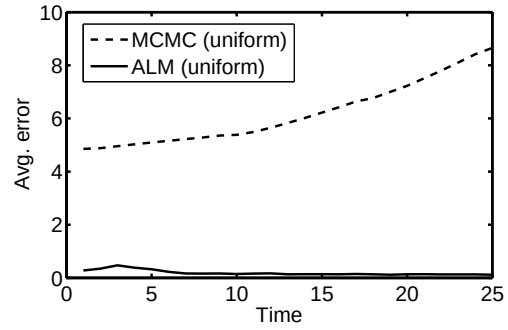


Figure 5. Average error (over 100 runs) of ALM and MCMC with uniform particle initialization (uniform) as a function of time. The results are shown for $N_{ppt} = 250$ and $M = 378$.

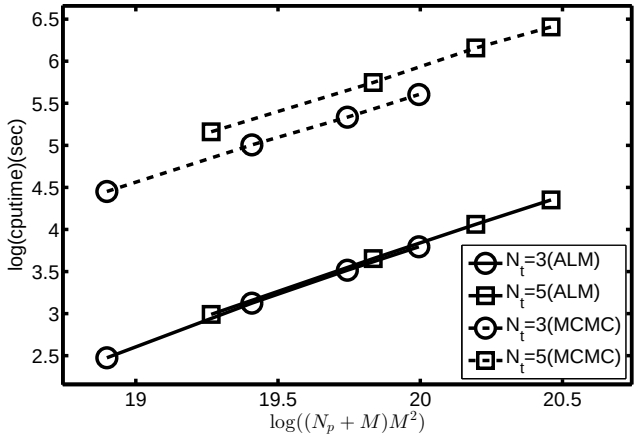


Figure 7. Computational complexity of ALM and MCMC as a function of theoretical complexity of ALM. Results shown for $M = 378$ and $N_{ppt} = [250, 500, 750, 1000]$.

In Fig. 6, particles are initialized in the vicinity of actual target state (init). This results in a much lower average error and it does not increase as the tracking progresses. In both scenarios, the average error for ALM is slightly higher than the MCMC (init) method in the first few iterations, but eventually decrease to an average error below 0.2.

In Fig. 7, we show the average CPU time (computational complexity) of ALM and MCMC for the same computer configuration[2] as a function of $(N_p+M)M^2$ (theoretical complexity of the ALM approach). The four values of each curve are four different values of $N_{ppt} = [250, 500, 750, 1000]$ and $M = 378$ is constant. The closely overlapping linear curves for the ALM algorithm for different target numbers indicate that the empirical computational time is accurately approximated

[2]Two Xeon 4-core 2.5GHz, 14GB RAM.

by the theoretical complexity. The complexity of the MCMC approach is $\mathcal{O}(N_t M N_p)$, but the empirical computational time is higher than the ALM algorithm. This is because the hidden constants within the complexity computations are much higher for MCMC method and the algorithm is much slower.

Table I
CPU TIME PER ITERATION IN SECONDS (AVERAGED OVER 100 RUNS) WITH STANDARD DEVIATION FOR $N_{ppt} = 250$ AND $M = 378$.

	$N_o = 3$	$N_o = 5$
ALM	0.47 ± 0.01	0.80 ± 0.01
MCMC	3.4 ± 0.2	7.0 ± 0.6

In Tab. I, we show the average computational time per iteration (with standard deviation) of both ALM and MCMC for 100 runs with $N_{ppt} = 250$ and $M = 378$. The ALM approach provides gains greater than 85% in both scenarios.

VII. CONCLUSION

In this paper, we investigated the problem of tracking multiple targets in an environment where each measurement is affected by multiple targets and each target contributes to multiple measurements. As a motivating example, we considered RF tomographic tracking. We derived the equations of a novel moment-based filter (ALM) for multi-target tracking scenarios where the likelihood model can be expressed as a function of the sum of the individual target contributions. We also considered the special case where the measurement model is Gaussian to further reduce the update equation. We performed Matlab simulations and compared ALM with an MCMC algorithm. We showed that we can achieve comparable error with 85% savings in computational time.

In the future, we would like to use active learning strategies to choose only a subset of sensors at each iteration to perform measurements. This will (i) enhance the life of the sensors and (ii) reduce the complexity of the ALM algorithm, which depends directly on the number of measurements M. This will also allow us to establish a relationship between the density of targets in the monitoring area and the minimum sensor density required to achieve a specified level of error.

APPENDIX

The updated PHD can be expressed as:

$$D_{k+1|k+1}(\mathbf{x}) = \int f_{k+1|k+1}(\{\mathbf{x}\} \cup W | Z^{(k+1)}) \delta W \quad (25)$$

Given the specific structure of our likelihood (see (10)), we can then write

$$\begin{aligned} &D_{k+1|k+1}(\mathbf{x}) \\ &= K^{-1} \int f_{k+1}(Z_{k+1}|\{\mathbf{x}\} \cup W) f_{k+1|k}(\{\mathbf{x}\} \cup W | Z^{(k)}) \delta W \\ &= K^{-1} \int f_{k+1}(Z'_{k+1}|W) f_{k+1|k}(\{\mathbf{x}\} \cup W | Z^{(k)}) \delta W \quad (26) \end{aligned}$$

where $Z'_{k+1} = Z_{k+1} - g(\mathbf{x})$ and K is a normalizing constant. Assume that $f_{k+1|k}(\{\mathbf{x}\} \cup W | Z^{(k)})$ is a Poisson point process, i.e. for $W = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, we can write:

$$f_{k+1|k}(\{\mathbf{x}\} \cup W | Z^{(k+1)}) = e^{-\lambda} I(\mathbf{x}) I(\mathbf{x}_1) \ldots I(\mathbf{x}_n) \quad (27)$$

for a mean rate λ and local intensity values $I(\mathbf{x})$. Then:

$$\begin{aligned} &D_{k+1|k+1}(\mathbf{x}) \\ &= K^{-1} I(\mathbf{x}) \int f_{k+1}(Z'_{k+1}|W) e^{-\lambda} I(\mathbf{x}_1) \ldots I(\mathbf{x}_n) \delta W \quad (28) \\ &= K^{-1} D_{k+1|k}(\mathbf{x}) \int f_{k+1}(Z'_{k+1}|W) f_{k+1|k}(W|Z^{(k)}) \delta W \quad (29) \end{aligned}$$

where we have used the result that $D_{k+1|k}(\mathbf{x}) = I(\mathbf{x})$ for a Poisson point process. Define the normalizing constant K as:

$$K = \int f_{k+1}(Z_{k+1}|W) f_{k+1|k}(W|Z^{(k)}) \delta W \quad (30)$$

Hence we can then write (29) as

$$D_{k+1|k+1}(\mathbf{x}) = \frac{D_{k+1|k}(\mathbf{x})}{F_{k+1}(\mathbf{x})} \quad (31)$$

where

$$F_{k+1}(\mathbf{x}) = \frac{\int f_{k+1}(Z_{k+1}|W) f_{k+1|k}(W|Z^{(k)}) \delta W}{\int f_{k+1}(Z'_{k+1}|W) f_{k+1|k}(W|Z^{(k)}) \delta W} \quad (32)$$

Let $P(s) = \Pr\left(\sum_{\mathbf{x}_i \in W} g_k(\mathbf{x}_i) = s\right)$ for the Poisson process described by $f_{k+1|k}(W|Z^{(k)})$ and g_k. Then we have:

$$F_{k+1}(\mathbf{x}) = \frac{\int_0^\infty h_Z(s) P(s) ds}{\int_0^\infty h_{Z'}(s) P(s) ds} \quad (33)$$

Using Campbell's theorem [15] we can analyze the distribution of a summation of a real-valued function defined over the state space of the Poisson process. It can be applied if:

$$\int_S \min(|g_k(\mathbf{x})|, 1) \lambda(\mathbf{x}) d\mathbf{x} < \infty \quad (34)$$

where S is the state-space over which the Poisson process is defined and λ is the intensity. This conditions holds for all $g_k(\mathbf{x})$ because $\lambda(x)$ is integrable - it (should) integrate to the number of targets. The theorem then states that we can identify the characteristic (moment generating) function of s as:

$$\mathbb{E}(e^{\theta s}) = \exp\left\{ \int_S (e^{\theta g_k(\mathbf{x})} - 1) \lambda(\mathbf{x}) d\mathbf{x} \right\}. \quad (35)$$

This leads to the following expressions for the mean and variance of s:

$$\mathbb{E}(s) = \int_S g_k(\mathbf{x}) \lambda(\mathbf{x}) \, d\mathbf{x} \quad (36)$$

$$\mathrm{var}(s) = \int_S g_k(\mathbf{x})^2 \lambda(\mathbf{x}) \, d\mathbf{x} \quad (37)$$

Furthermore, as long as

$$\int_S g_k^j(\mathbf{x})^2 \lambda(\mathbf{x}) d\mathbf{x} < \infty \quad (38)$$

holds[3] for every $g_k^j(\mathbf{x})$, then:

$$\mathrm{cov}(s_a, s_b) = \int_S g_k^a(\mathbf{x}) g_k^b(\mathbf{x}) \lambda(\mathbf{x}) d\mathbf{x} \tag{39}$$

We adopt a Gaussian approximation to $P(s)$, setting $P(s) = \mathcal{N}_s(\mu_s, \Sigma^2)$ where the mean and variance are identified by the values above. Therefore, we can write

$$F_{k+1}(\mathbf{x}) \approx \frac{\int_0^\infty h_Z(s) \mathcal{N}_s(\mu_s, \Sigma) ds}{\int_0^\infty h_{Z'}(s) \mathcal{N}_s(\mu_s, \Sigma) ds} \tag{40}$$

where

$$\mathcal{N}_x(\mu, \Sigma) = \exp\left[-\frac{1}{2}(x-\mu)\Sigma^{-1}(x-\mu)\right] \tag{41}$$

When the measurement model is also Gaussian, we can further reduce the expression for $F_{k+1}(\mathbf{x})$. Let $h_{Z'}(s) = \mathcal{N}_{Z'}(s, \Sigma_s)$. Then we can write

$$\begin{aligned}\int_0^\infty h_{Z'}(s) \mathcal{N}_s(\mu_s, \Sigma) ds &= \int_0^\infty \mathcal{N}_s(Z'_k, \Sigma) \mathcal{N}_s(\mu_s, \Sigma) ds \\ &\approx C \cdot \mathcal{N}_{Z_k}(g(\mathbf{x}) + \mu_s, \Sigma_s + \Sigma)\end{aligned} \tag{42}$$

where C is a constant that does not need to be computed in our case because it appears at both the numerator and denominator of (40). Note that to derive an analytical expression for this integral, we need to make the approximation that $\int_0^\infty \cdot ds \approx \int_{-\infty}^\infty \cdot ds$, which introduces an additional error. The entire expression for $F_{k+1}(\mathbf{x})$ can now be approximated as:

$$F_{k+1}(\mathbf{x}) \approx \frac{\mathcal{N}_{Z_k}(\mu_s, \Sigma_s + \Sigma)}{\mathcal{N}_{Z_k}(g(\mathbf{x}) + \mu_s, \Sigma_s + \Sigma)} \tag{43}$$

REFERENCES

[1] Y. Ho and R. Lee, "A Bayesian approach to problems in stochastic estimation and control," *IEEE Trans. on Automatic Control*, vol. 9, no. 4, pp. 333–339, Oct. 1964.

[2] R. Mahler, "Multitarget Bayes filtering via first-order multitarget moments," *IEEE Trans. on Aerospace and Electronic Systems,*, vol. 39, no. 4, pp. 1152–1178, Oct. 2003.

[3] B. Balakumar, A. Sinha, T. Kirubarajan, and J. Reilly, "PHD filtering for tracking an unknown number of sources using an array of sensors," in *Proc. IEEE/SP Work. on Statistical Signal Processing*, Bordeaux, France, Jul. 2005.

[4] D. Angelosante, E. Biglieri, and M. Lops, "Multiuser detection in a dynamic environment: Joint user identification and parameter estimation," in *Proc. IEEE Int. Symp. on Information Theory*, Nice, France, Jun. 2007.

[5] J. Wilson and N. Patwari, "Radio tomographic imaging with wireless networks," *IEEE Trans. Mobile Computing*, vol. 9, no. 5, pp. 621–632, Jan. 2010.

[6] R. Mahler, *Statistical multisource-multitarget information fusion*. Artech House, 2007.

[7] F. Septier, S. Pang, A. Carmi, and S. Godsill, "On MCMC-based particle methods for Bayesian filtering: Application to multitarget tracking," in *Proc. IEEE Int. Work. Computational Advances in Multi-Sensor Adaptive Processing*, Aruba, Dutch Antilles, Dec. 2009.

[8] Z. Khan, T. Balch, and F. Dellaert, "MCMC-based particle filtering for tracking a variable number of interacting targets," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, pp. 1805–1918, Nov. 2005.

[9] S. K. Pang, J. Li, and S. Godsill, "Models and algorithms for detection and tracking of coordinated groups," in *Proc. IEEE Aerospace Conf.*, Big Sky, MT, Mar. 2008.

[3]Given the structure of $g_k(\mathbf{x})$ this condition is clearly satisfied.

[10] H. Sidenbladh, "Multi-target particle filtering for the probability hypothesis density," in *Proc. Int. Conf. Information Fusion*, Cairns, Australia, Jul. 2003.

[11] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo methods for multitarget filtering with random finite sets," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1224–1245, Oct. 2005.

[12] M. Fanbin, H. Yanling, X. Quanxi, O. Taishan, and Z. Wei, "A particle PHD filter for multi-sensor multi-target tracking based on sequential fusion," in *Proc. Int. Conf. on Information Engineering and Computer Science*, Wuhan, China, Dec. 2009.

[13] R. Mahler, "CPHD filters for superpositional sensors," in *Proc. Signal and Data Processing of Small Targets*, San Diego, CA, Aug. 2009.

[14] X. Chen, A. Edelstein, Y. Li, M. Coates, A. Men, and M. Rabbat, "Sequential Monte Carlo for simultaneous passive device-free tracking and sensor localization using received signal strength measurements," in *Proc. IEEE/ACM Int. Conf. on Information Processing in Sensor Networks*, Chicago, IL, Apr. 2011.

[15] J. Kingman, *Poisson Processes*. Oxford, UK: Clarendon Press, 1993.

[16] J. Evans and D. Everitt, "On the teletraffic capacity of CDMA cellular networks," *IEEE Trans. on Vehicular Technology*, vol. 48, no. 1, pp. 153–165, Jan. 1999.

[17] C. C. Chan and S. Hanly, "Calculating the outage probability in a CDMA network with spatial poisson traffic," *IEEE Trans. on Vehicular Technology*, vol. 50, no. 1, pp. 183–204, Jan. 2001.

[18] D. Clark, I. Ruiz, Y. Petillot, and J. Bell, "Particle PHD filter multiple target tracking in sonar image," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 43, no. 4, pp. 1441–1453, Oct. 2007.

[19] Y. Li, X. Chen, M. Coates, and B. Yang, "Sequential Monte Carlo radio-frequency tomographic tracking," 2011, to appear in Proc. Int. Conf. on Acoustics, Speech and Signal Processing.

[20] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Trans. on Signal Processing*, vol. 56, no. 8, pp. 3447–3457, Aug. 2008.