A Particle Marginal Metropolis-Hastings Multi-target Tracker

Anh-Tuyet Vu, member, *IEEE*, Ba-Ngu Vo, member, *IEEE*, and Rob Evans, Fellow, *IEEE*

***Abstract*—We propose a Bayesian multi-target batch processing algorithm capable of tracking an unknown number of targets that move close and/or cross each other in a dense clutter environment. The optimal Bayes multi-target tracking problem is formulated in the random finite set framework and a Particle Marginal Metropolis-Hastings (PMMH) technique which is a combination of Metropolis-Hastings (MH) algorithm and sequential Monte Carlo methods is applied to compute the multi-target posterior distribution. The PMMH technique is used to design a high dimensional proposal distributions for the MH algorithm and allows the proposed batch process multi-target tracker to handle a large number of tracks in a computationally feasible manner. Our simulations show that the proposed tracker reliably estimates the number of tracks and their trajectories in scenarios with a large number of closely spaced tracks in a dense clutter environment albeit, more expensive than on-line methods.**

Keywords: Multi-target Tracking, Particle Marginal Metropolis-Hastings, Metropolis-Hastings, Markov Chain Monte Carlo, Random Sets, Sequential Monte Carlo.

I. Introduction

Multi-target tracking (MTT) deals with the problem of estimating sequences of the states of unknown and time-varying number of targets based on sequence of noisy and clutter measurement sets from a variety of sensors. Over the past 40 years MTT has emerged as a fundamentally important technology with diverse applications [1], [2] such as radar tracking for aircrafts, passive and active acoustic tracking for seacrafts; video tracking of people for security applications; tracking directions of arrival (DOAs) of an unknown number of sources, using a passive array of sensors [3], etc.

The MTT problem is characterized by $1)$ the targets being tracked may randomly appear and disappear from the field of view; and $2)$ they may be temporarily obscured by other objects; Furthermore, the sensor measurements suffer from imperfections such as $3)$ noise corruption which introduced location errors and may cause missed detection of objects; $4)$ the existence of false measurements which do not belong to a valid object of interest which may lead to misidentification. The problem we investigate not only inherits the four aforementioned challenges but also has the properties that $5)$ targets may merge and split, and may cross or travel very close to each other for extended periods of time and $6)$ there are a large numbers of targets moving in a dense clutter environment. The majority of traditional MTT algorithms such as Multiple Hypothesis Tracking (MHT) [4], Joint Probabilistic Data Association (JPDA) [5], Joint Integrated Probabilistic Data Association - JIPDA [6], and their variants [7] have been found to perform effectively provided the density of targets and the number of false measurements are modest by hypothesizing the association between the measurements and targets. However, these techniques are no longer adequate when the density of targets is high and the number of false measurements is large.

In general, the number of unknown targets in MTT problems also needs to be estimated along with their states so the random finite set (RFS) framework provides a more natural treatment than the traditional ones. Since the introduction of the RFS based techniques [8]–[10], they have been attracting many researchers around the world due to their capabilities to avoid data association and to capture $1)$ the linear and non-linear model and $2)$ the six properties of aforementioned problem. Modeling the MTT problem in the RFS framework allows the full multi-target Bayesian filter to propagate the full multi-target posterior distribution in a similar way to the single-target Bayesian filter taking into account aforementioned properties. An on-line solution to the full Bayes multi-target filter have been proposed in [11]. Computationally efficient approximation have also been the subject to intensive investigation during the last decade. The Probability hypothesis density (PHD) filter [12], is introduced to propagate the first-order statistical moment (intensity) of the posterior distribution in time and operates on the single-target state space. At each time t the PHD filter estimates the number of targets N_t by integrating the PHD (intensity) over the spatial region of interest, rounding N_t to the nearest integer $\hat{N}_t$ and then selecting $\hat{N}_t$ local maximal corresponding to $\hat{N}_t$ target states. Its implementations [13]–[15] using Gaussian mixture methods and sequential Monte Carlo (SMC) were derived to track targets and its number under linear-Gaussian or mildly non-linear target dynamic assumptions. The PHD filter and its variants are justifiable provided the signal to noise ratio (SNR) of a single target is high enough to remediate the information loss from the approximation of the full multi-target Bayesian filter and the targets are not so close. As stated by [16], the PHD filter did not give a reliable solution of the target numbers due to missed detection and/or significant large clutter density. The cardinality probability hypothesis density (CPHD) filter, a generalization of PHD filter [17], was introduced to overcome this issue by propagating not only the first-order moment

Anh-Tuyet Vu (email: vutta@unimelb.edu.au) is with the Department of Electrical and Electronic Engineering, the University of Melbourne, Vic. 3010, Australia.

B. Vo (e-mail: ba-ngu.vo@curtin.edu. au) is with Department of Electrical and Computer Engineering, Curtin University, Bentley, WA 6102, Australia .

Rob Evans (email: robinje@unimelb.edu.au) is with the Department of Electrical and Electronic Engineering, the University of Melbourne, Vic. 3010, Australia.

of the posterior distribution in time but also the probability distribution of target number and its probability generating function. The CPHD filter were implemented using Gaussian mixture [18] and showed more accurate and stable target number estimates than Gaussian mixture PHD (GM-PHD) filter. However, the computational complexity ($\geq \mathcal{O}(n_t^3 m_t)$) is much higher than the PHD filter ($\geq \mathcal{O}(n_t m_t)$) where n_t is the number of measurements and m_t is the number of detected targets. Similar to the PHD filter, the CPHD filter does not explicitly produce tracks without some post processing.

The main contribution of this paper is the development of a Bayesian multi-target batch processing algorithm based on RFS modeling and a Particle Marginal Metropolis-Hastings (PMMH) numerical approximation [19] using an estimate from a GM-PHD Tracker (GM-PHDT) [15] as initialization. The multi-target Bayesian recursion, which involves multiple integrals of a sequence of sets, can be solved by sampling methods. The proposed technique involves three steps: 1) Derive the posterior distribution of trajectories of multiple targets conditional on all measurements; 2) Find independent samples from this posterior distribution where each sample is a sequence of multi-target states over time; and 3) Estimate the trajectories of multiple targets based on the maximum likelihood principles. The initial result of this work has been published in [20]. Most current techniques only consider a problem with at most the first 5 properties of the aforementioned problem. The proposed algorithm is derived to deal with all these 6 properties by sampling directly from the RFS based Bayesian multi-target posterior distribution so it avoids information loss opposed to PHD or CPHD which approximate the posterior distribution. Moreover, in order to avoid a large number of data association, it only considers one hypothesis concerning the relationship between targets and measurements at a time compared to the MHT approach which in principle, considers all possible hypotheses between targets and measurements.

The proposed algorithm is called the PMMH Multi-target tracker (PMMH-MTT). As a part of the PMMH-MTT, we design a Markov chain (MC) which has the desired distribution as the stationary distribution. The initial states of the MC is chosen as an estimate from the Gaussian Mixture Probability Hypothesis Density Tracker (GM-PHDT) [15] in order to reduce significantly the computation time or the burn-in time of the MC. The capabilities of the proposed algorithm are illustrated in simulations. We show that the algorithm is able to estimate reliable trajectories of the targets in a scenario with closely spaced and crossing tracks immersed in dense clutter. Moreover, we demonstrate that it can separate targets which cross each other or close to each other for a long period.

The structure of this paper is as follows. Section II describes a Bayesian formulation of MTT problem using RFS framework. The proposed tracker, PMMH-MTT, is presented in Section III. Section IV presents the numerical simulations and discussion.

II. Problem formulation

This section presents a Bayesian formulation of the multi-target tracking problem. Subsection II-A describes the multi-target system model and subsection II-B is devoted to the problem formulation in a RFS framework.

Notation: $u_{1:t} = (u_1, \ldots, u_t)$ denotes a sequence with elements $u_1, \ldots, u_t$ where $u_j, j = 1, \ldots, t$ is a vector or a set. $|A|$ denotes the cardinality of the finite set A. Let $m > n$, denote $C_m^n = \frac{m!}{n!(m-n)!}$. For notational consistency, we use a capital letter for a set and a lower case letter for a scalar or vector. Also, a bold capital letter denotes a function whose output is a set, and a bold lower case letter denotes a function whose output is a vector or scalar. We make an exception for ω and θ because we want to use the same notation as in [21] and [10, Chapter 10.5.4, p.332].

A. Random Finite Set Model

Let $\mathcal{T} = \{1, \ldots, T\}$ be the set of time indices where T is the number of instances where measurements are taken. In the RFS framework, a multi-target state and a multi-target measurement at time t are respectively represented as finite sets X_t and Z_t. If m_t targets are present at time t, the multi-target state is $X_t = \{x_1, x_2, \ldots, x_{m_t}\} \subset \mathcal{X}$ where $\mathcal{X} \subseteq \mathbb{R}^{n_x}$ is the single-target state space and n_x is the dimension of the single-target state space. Similarly, if there are n_t observations at time t, the multi-target observation is $Z_t = \{z_1, \ldots, z_{n_t}\} \subset \mathcal{Z}$ where $\mathcal{Z} \subseteq \mathbb{R}^{n_z}$ is the measurement space and n_z is the dimension of the single-target measurement space.

Given a single-target state $x_{t-1} \in X_{t-1}$ at time $t-1$, its behavior at time t is modeled by the Bernoulli RFS $\mathbf{S}_{t|t-1}(x_{t-1})$ that is either $\{x_t\}$ according to probability density $f(x_t|x_{t-1})$ when the target survives with the probability $p_{S_t}(x_{t-1})$ or $\emptyset$ when the target dies with the probability $1 - p_{S_t}(x_{t-1})$. Thus the set of the survival and death of all existing targets from time $t-1$ to time t is modeled by

$$\boldsymbol{\mathcal{S}}_{t|t-1}(X_{t-1}) = \bigcup_{x \in X_{t-1}} \mathbf{S}_{t|t-1}(x).$$

In addition to the set of the surviving targets, there is a set of spawned targets, denoted by $\boldsymbol{\mathcal{B}}_{\boldsymbol{t|t-1}}(X_{t-1})$ and a set of spontaneous births, denoted by Γ_t such that

$$X_t = \boldsymbol{\mathcal{S}}_{t|t-1}(X_{t-1}) \cup \boldsymbol{\mathcal{B}}_{\boldsymbol{t|t-1}}(X_{t-1}) \cup \Gamma_t.$$

The set $\boldsymbol{\mathcal{S}}_{t|t-1}(X_{t-1})$, $\boldsymbol{\mathcal{B}}_{\boldsymbol{t|t-1}}(X_{t-1})$ and Γ_t are conditionally independent (given X_{t-1}). Let $\pi_{\boldsymbol{\mathcal{S}},t|t-1}(\cdot|X_{t-1})$, $\pi_{\boldsymbol{\mathcal{B}},t|t-1}(\cdot|X_{t-1})$ and $\pi_{\Gamma,t}(\cdot)$ be the probability densities of the RFS of survival $\boldsymbol{\mathcal{S}}_{t|t-1}(X_{t-1})$, spawning $\boldsymbol{\mathcal{B}}_{t|t-1}(X_{t-1})$ and spontaneous birth Γ_t respectively. Then the multi-target state transition can be described as a *multi-target transition density*

$$f_{t|t-1}(X_t|X_{t-1}) = \sum_{U \subseteq X_t} \pi_{\boldsymbol{\mathcal{S}},t|t-1}(U|X_{t-1}) \sum_{V \subseteq X_t - U} \pi_{\boldsymbol{\mathcal{B}},t|t-1}(V|X_{t-1}) \times \pi_{\Gamma,t}(X_t - U - V) \quad (1)$$

where $f_{1|0}(X_1|X_0) = \mu_0(X_1)$.

In order to distinguish multiple target trajectories we augment each single-target state with a target label. Thus our augmented single-target state space is

$$\widetilde{\mathcal{X}} = \mathcal{X} \times \mathcal{K}. \quad (2)$$

where $\mathcal{K} = \{1, \cdots, K\}$ is a set of target labels and K is the maximum number of targets which is assumed known.

Let $\tilde{x}_{t-1} = (x_{t-1}, k)$ and $\tilde{x}_t = (x_t, k')$, then the augmented single-target transition density $\bar{f}_{t|t-1}(\tilde{x}_t|\tilde{x}_{t-1})$ is given by

$$\bar{f}_{t|t-1}(\tilde{x}_t|\tilde{x}_{t-1}) = \begin{cases} \bar{f}(x_t|x_{t-1}), & \text{if } k = k'; \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

Let $\boldsymbol{\mathcal{L}}(\widetilde{X}_{t-1})$ be a set of target labels of $\widetilde{X}_{t-1}$. The set of surviving targets is denoted by $W^* = \{(x,k) \in \widetilde{X}_t : k \in \boldsymbol{\mathcal{L}}(\widetilde{X}_{t-1})\}$ and its transition density is the third product of (4). Let $\mathrm{T}(U,V)$ denote the set of all $1-1$ functions from a finite set U to a finite set V. Then $\alpha \in \mathrm{T}(W^*, \widetilde{X}_{t-1})$ associates targets at time t with targets at time $t-1$. Specifically, $(x_{t-1}, k) = \alpha(x_t, k')$ if $k = k'$, i.e. the target state x_{t-1} at time $t-1$ has evolved to target state x_t at time t. Since the survival of a target $\tilde{x}_{t-1} = (x_{t-1}, k)$ is independent of its label, we write $p_{S_t}(\tilde{x}_{t-1}) = p_{S_t}(x_{t-1})$. Target state $\tilde{x}_{t-1}$ at time $t-1$ not associated with any target state at time t is dead. The set of dead targets is $\widetilde{X}_{t-1} - \alpha(W^*)$ and their probabilities are the second product of (4). Given $\widetilde{X}_t$ and $\widetilde{X}_{t-1}$ we know the target labels of $\widetilde{X}_t$ and $\widetilde{X}_{t-1}$ and hence we know which target survives, dies or is new (spawning or birth).

Assume that the number of new targets follows a Poisson distribution with mean $\mu_f(X_{t-1})$ (Since the appearance of a new target is independent of target label, we denote $\mu_f(\widetilde{X}_{t-1}) = \mu_f(X_{t-1})$); and that $b(\tilde{x}|\widetilde{X}_{t-1})$ is the intensity of a new target $\tilde{x} = (x,k)$[1] $\in \widetilde{X}_t$. The set of new targets is $\widetilde{X}_t - W^*$ and its transition density is the first term and the first product of (4). The transition density $f_{t|t-1}(\widetilde{X}_t|\widetilde{X}_{t-1})$ is(see [22, Chapter 3.2.1, p.44-48 and Chapter 6.2.1.1, p.104] for details)

$$\begin{aligned} f_{t|t-1}(\widetilde{X}_t|\widetilde{X}_{t-1}) =& e^{-\mu_f(\widetilde{X}_{t-1})} \frac{1}{|\widetilde{X}_t - W^*|!} \prod_{\tilde{x} \in \widetilde{X}_t - W^*} b(\tilde{x}|\widetilde{X}_{t-1}) \\ & \times \prod_{\tilde{x} \in \widetilde{X}_{t-1} - \alpha(W^*) : \alpha \in \mathrm{T}(W^*, \widetilde{X}_{t-1})} (1 - p_{S_t}(\tilde{x})) \\ & \times \left(\prod_{\tilde{x} \in W^*} p_{S_t}(\alpha(\tilde{x})) \bar{f}_{t|t-1}(\tilde{x}|\alpha(\tilde{x})) \right). \end{aligned} \quad (4)$$

At time t, each single-target state x_t of $\tilde{x}_t$ is either detected with probability $p_{D_t}(x_t)$ and generates measurement z_t with likelihood $\bar{g}_t(z_t|x_t)$ or is missed with probability $1 - p_{D_t}(x_t)$. Thus a target generates an RFS $\boldsymbol{D}_t(x_t)$ which can take either the value $\{z_t\}$ with likelihood $\bar{g}_t(z_t|x_t)$ or $\emptyset$. The set of target-generated measurements is

$$\boldsymbol{\mathcal{D}}_t(X_t) = \bigcup_{x \in X_t} \boldsymbol{D}_t(x).$$

Apart from target-originated measurements, the sensor also receives a set of false/spurious measurements or clutter which is modeled by an RFS Λ_t. Thus, Z_t, the measurement set at time t, is the union of the target-generated measurement set and the clutter set, hence

$$Z_t = \boldsymbol{\mathcal{D}}_t(X_t) \cup \Lambda_t.$$

[1]In $\tilde{x} = (x,k)$ only x is stochastic, k is assigned the smallest available target label.

Given the probability density $\pi_{\boldsymbol{\mathcal{D}},t}(\cdot|\cdot)$ of target-generated measurement $\boldsymbol{\mathcal{D}}_t(\cdot)$ and the probability density $\pi_{\Lambda,t}(\cdot)$ of clutter Λ_t, the augmented multi-target likelihood is given by

$$g_t(Z_t|\widetilde{X}_t) = g_t(Z_t|X_t) = \sum_{U \subseteq Z_t} \pi_{\boldsymbol{\mathcal{D}},t}(U|X_t) \pi_{\Lambda,t}(Z_t - U). \quad (5)$$

Since the target-generated measurements is independent of target labels, $g_t(Z_t|\widetilde{X}_t) = g_t(Z_t|X_t)$ and $\bar{g}_t(z|\tilde{x}) = \bar{g}_t(z|x)$ where $\tilde{x} \in \widetilde{X}_t$. We can use $p_{D_t}(\tilde{x})$ instead of $p_{D_t}(x)$ since the probability of detection is independent of target label. Assuming that the clutter RFS is Poisson with intensity κ_t, then (5) is expanded as follows

$$\begin{aligned} g_t(Z_t|\widetilde{X}_t) = & \sum_{W \subseteq Z_t} \sum_{\alpha \in \mathrm{T}(W, \widetilde{X}_t)} \prod_{z \in W} p_{D_t}(\alpha(z)) \bar{g}_t(z|\alpha(z)) \times \\ & \left(e^{-\langle \kappa_t, 1 \rangle} \prod_{z \in Z_t - W} \kappa_t(z) \right) \prod_{\tilde{x} \in \widetilde{X}_t - \alpha(W)} (1 - p_{D_t}(\tilde{x})). \end{aligned} \quad (6)$$

where $\langle u, v \rangle = \int u(x) v(x) dx$; and $\mathrm{T}(W, \widetilde{X}_t) = \emptyset$ if $|W| > |\widetilde{X}_t|$ and the sum over the empty set is zero. The first product in (6) is the density of measurements corresponding to detected targets, the second product and $e^{-\langle \kappa_t, 1 \rangle}$ is the density of clutter; the last product is the probabilities of undetected targets.

For notational simplicity, $f(\widetilde{X}_t|\widetilde{X}_{t-1})$ and $g(Z_t|\widetilde{X}_t)$ are used in place of $f_{t|t-1}(\widetilde{X}_t|\widetilde{X}_{t-1})$ and $g_t(Z_t|\widetilde{X}_t)$ respectively if there is no ambiguity.

B. Track in the RFS framework

In this section, we define the notion of a track hypothesis which is a particular collection of tracks. A track (trajectory of a target) in a track hypothesis is a collection of at least m^* single-target states at consecutive times with the same label. The track gate m^* is a minimum number of states a track must contain in order to be called a track. A track is defined as follows

Definition 1 (*Track*) *Given a track gate m^*, a track τ is an array of the form*

$$\tau = (k, t, x_0, \ldots, x_m), \qquad m \geq m^* - 1 \quad (7)$$

where $k \in \mathcal{K}$ is the track label, $t \in \mathcal{T}$ is the initial time of the track, $x_i \in \mathcal{X}$ is state of the track at time $t+i$ for $i = 0, \ldots, m$.

For the track τ in (7)*, we denote the instances of track existence, the initial time of the track, and the track label respectively by*

$$\mathfrak{T}(\tau) = \{t, t+1, \ldots, t+m\}, \qquad \mathbf{t}_0(\tau) = t, \qquad \boldsymbol{l}(\tau) = k.$$

For $t' \in \mathfrak{T}(\tau)$, we denote the state at time t' and the augmented state at time t' respectively by

$$\mathbf{x}_{t'}(\tau) = x_{t'-t}, \qquad \tilde{\mathbf{x}}_{t'}(\tau) = (x_{t'-t}, k).$$

Similarly, given an augmented single-target state $\tilde{x} = (x,k)$ we also denote the label of $\tilde{x}$ by $\boldsymbol{l}(\tilde{x}) = k$. We are only interested in a special collection of tracks which represents the

estimates of the trajectories of underlying targets and which is defined next.

Definition 2 (*Track Hypothesis*) *A Track Hypothesis ω is a set of tracks such that no two tracks share the same label and no two tracks share the same state at the same time i.e. for all $\tau, \tau' \in \omega$*

1) $\boldsymbol{l}(\tau) \neq \boldsymbol{l}(\tau')$ *and*
2) $\mathbf{x}_t(\tau) \neq \mathbf{x}_t(\tau')$ *for any* $t \in \mathfrak{T}(\tau) \cap \mathfrak{T}(\tau')$.

For a Track Hypothesis ω, we denote the set of augmented multi-target states at time $t \in \mathcal{T}$ by

$$\widetilde{\mathbf{X}}_t(\omega) \quad = \quad \{\tilde{\mathbf{x}}_t(\tau) : \tau \in \omega\}.$$

One of our objective in this paper is to sample the *Track Hypothesis* ω from the posterior distribution $p(\omega|Z_{1:T})$. Note that ω can be equivalently specified by an array of augmented multi-target states $\widetilde{\mathbf{X}}_{1:T}(\omega) = (\widetilde{\mathbf{X}}_1(\omega), \ldots, \widetilde{\mathbf{X}}_T(\omega))$. Let $\widetilde{X}_t = \widetilde{\mathbf{X}}_t(\omega)$ for $t = 1, \ldots, T$, then $p(\omega|Z_{1:T})$ of ω conditional on $Z_{1:T}$ is (detail in [22, Chapter 6.23, p.108])

$$p(\omega|Z_{1:T}) = p(\widetilde{X}_{1:T}|Z_{1:T}) = \frac{\prod_{t=1}^{T} f(\widetilde{X}_t|\widetilde{X}_{t-1}) g(Z_t|\widetilde{X}_t)}{p(Z_{1:T})}. \quad (8)$$

The multi-target posterior distribution in (8) is computationally intractable for a large number of tracks in a dense environment because the computations of the likelihood function $g(Z_t|\widetilde{X}_t)$ in (8) involves all possible combinations between the target states and measurements. Our objective is to find a numerical approximation of the posterior distribution of ω. Later on we will use the PMMH sampler to find such an approximation and we next rewrite (8) in a form convenient for this purpose.

Let $n_t \in \{0\} \cup \mathbb{N}$, $B = \{0, \ldots, n_t\}, A \in \{U : U \subset \mathbb{N} \text{ or } U = \emptyset\}$. Then define an auxiliary function θ_t^{A,n_t} as a mapping from A to B with the property that

$$\theta_t^{A,n_t}(k) = \theta_t^{A,n_t}(k') > 0 \text{ implies that } k = k'. \quad (9)$$

Thus, the probability $p(\theta_t^{A,n_t})$ is naturally defined by

$$p(\theta_t^{A,n_t}) = \frac{1}{\sum_{i=0}^{\min\{|A|,n_t\}} i! C_{|A|}^{i} C_{n_t}^{i}}. \quad (10)$$

Let Θ_t^{A,n_t} be the collection of all functions θ_t^{A,n_t} with the property (9). It is easy to show that $p(\theta_t^{A,n_t})$ given in (10) is a probability measure on the function space Θ_t^{A,n_t}.

If n_t is the number of elements in Z_t, $z_{1:n_t}$ is a random ordering of the elements, and if A is the set of target labels at time t (i.e. $A = \boldsymbol{\mathcal{L}}(\widetilde{X}_t)$), then the auxiliary function $\theta_t^{\boldsymbol{\mathcal{L}}(\widetilde{X}_t),n_t}$ assigns target labels to measurement indices, e.g. $\theta_t^{\boldsymbol{\mathcal{L}}(\widetilde{X}_t),n_t}(k) = j$. If $j > 0$, a target with label k has generated the measurement z_j otherwise the target is not detected. We also require that no two targets are assigned the same measurement index. If this happens one of the targets will be regarded as undetected.

Given $\theta_t^{\boldsymbol{\mathcal{L}}(\widetilde{X}_t),n_t}$, in addition to the association between targets and target-generated measurements, we also know which targets are undetected and which measurement are clutter. Employing a form of the multi-target likelihood given in [10, Chapter 10.5.4, p.332], (6) can be written as

$$g(Z_t|\widetilde{X}_t) = \sum_{\theta_t^{\boldsymbol{\mathcal{L}}(\widetilde{X}_t),n_t}} g(z_{1:n_t}, \theta_t^{\boldsymbol{\mathcal{L}}(\widetilde{X}_t),n_t}|\widetilde{X}_t). \quad (11)$$

By (6) and (11), $g(z_{1:n_t}, \theta_t^{\boldsymbol{\mathcal{L}}(\widetilde{X}_t),n_t}|\widetilde{X}_t)$ is

$$\prod_{\tilde{x}\in\widetilde{X}_t:\theta_t^{\boldsymbol{\mathcal{L}}(\widetilde{X}_t),n_t}(\boldsymbol{l}(\tilde{x}))>0} p_{D_t}(\tilde{x}) \bar{g}_t(z_{\theta_t^{\boldsymbol{\mathcal{L}}(\widetilde{X}_t),n_t}(\boldsymbol{l}(\tilde{x}))}|\tilde{x})$$
$$\times e^{-\langle \kappa_t, 1\rangle} \prod_{i\notin\theta_t^{\boldsymbol{\mathcal{L}}(\widetilde{X}_t),n_t}} \kappa_t(z_i) \prod_{\tilde{x}\in\widetilde{X}_t:\theta_t^{\boldsymbol{\mathcal{L}}(\widetilde{X}_t),n_t}(\boldsymbol{l}(\tilde{x}))=0} (1 - p_{D_t}(\tilde{x})). \quad (12)$$

For each n_t and $|Z_t| = n_t$ (note that if $|Z_t| \neq n_t$, $g(z_{1:n_t}, \theta_t^{\boldsymbol{\mathcal{L}}(\widetilde{X}_t),n_t}|\widetilde{X}_t) = 0$), the summation in (11) is over all functions from $\boldsymbol{\mathcal{L}}(\widetilde{X}_t)$ to $\{0, \ldots, n_t\}$. $i \notin \theta_t^{\boldsymbol{\mathcal{L}}(\widetilde{X}_t),n_t}$ (i.e $i \notin \theta_t^{\boldsymbol{\mathcal{L}}(\widetilde{X}_t),n_t}(\boldsymbol{\mathcal{L}}(\widetilde{X}_t))$) are those measurement indices which are not associated with a target, i.e measurement indices not in the range of the function $\theta_t^{\boldsymbol{\mathcal{L}}(\widetilde{X}_t),n_t}$ is assumed clutter.

θ_t^{A,n_t} is extended to an augmented auxiliary function $\tilde{\theta}_t^{A,n_t}$

$$\tilde{\theta}_t^{A,n_t}(k) = (\theta_t^{A,n_t}(k), k). \quad (13)$$

Since k is deterministic, $p(\tilde{\theta}_t^{A,n_t}) = p(\theta_t^{A,n_t})$. For notational convenience, we drop the superscripts n_t and A. Note that $\tilde{\theta}_t, t \in \mathcal{T}$ are independent so $p(\tilde{\theta}_{1:T}) = \prod_{t=1}^{T} p(\tilde{\theta}_t)$ and $\tilde{\theta}_{1:T}$ represent all information about the targets apart from their locations. In fact, if $\theta_t(k) = 0$ for $t \leq t_0$ and $t \geq t_1$ then the target k appears at time t_0+1 and dies at time t_1. Moreover, we use Z, $\widetilde{Z}$, $\widetilde{X}$, $\tilde{\theta}$ in place of $Z_{1:T}$, $\widetilde{Z}_{1:T}$, $\widetilde{X}_{1:T}$, $\tilde{\theta}_{1:T}$ respectively if there is no ambiguity.

Now (8) takes the form

$$p(\omega|Z) = p(\widetilde{X}|Z) = \sum_{\tilde{\theta}} p(\widetilde{X}, \tilde{\theta}|\widetilde{Z}) \quad (14)$$

where $|Z_t| = n_t$, and $\tilde{\theta} = (\tilde{\theta}^{\boldsymbol{\mathcal{L}}(\widetilde{X}_1),n_1}, \ldots, \tilde{\theta}^{\boldsymbol{\mathcal{L}}(\widetilde{X}_T),n_T})$.

By Bayes rule we also have

$$p(\widetilde{X}, \tilde{\theta}|\widetilde{Z}) = p(\widetilde{X}|\widetilde{Z}, \tilde{\theta}) p(\tilde{\theta}|\widetilde{Z}). \quad (15)$$

Our aim is to sample $\tilde{\theta}$ and $\widetilde{X}$ from $p(\widetilde{X}, \tilde{\theta}|\widetilde{Z})$ in (14). The right hand side of (15) suggests that we can first sample $\tilde{\theta}$ from $p(\cdot|\widetilde{Z})$ and then sample $\widetilde{X}$ from $p(\cdot|\widetilde{Z}, \tilde{\theta})$. Based on this idea, the PMMH-MTT is presented in the section III. It uses a variant of SMC presented in subsection III-A to sample from $p(\widetilde{X}|\tilde{\theta}, \widetilde{Z})$ where $\tilde{\theta}$ is sampled from $p(\tilde{\theta}|\widetilde{Z})$ based on proposal moves [22] as discussed in subsection III-C.

From now on, we let $v_C(A|B)$ denote $v(A|B, C)$ where $C = \tilde{\theta}_{1:t}, t \in \mathcal{T}$ and v can be p, $\hat{p}$, q and q.

III. Particle Marginal Metropolis-Hastings - Multitarget tracker

This section presents the PMMH-MTT based on the PMMH sampler which exploits the strengths of SMC and Metropolis-Hastings (MH) approaches by combining these algorithms to sample from a high dimension probability distribution $p(\widetilde{X}, \tilde{\theta}|\widetilde{Z})$ that cannot be satisfactorily sampled by using only SMC or MH algorithms [19]. The PMMH algorithm,

an approximation of a Marginal metropolis-Hastings (MMH) algorithm, provides a flexible framework to carry out the inference by using particles obtained from SMC as a proposal density for an MH acceptance ratio. The MH acceptance ratio of $p(\widetilde{X}^*, \tilde{\theta}^*|\widetilde{Z})$ is of the form

$$\eta = \frac{p(\widetilde{X}^*, \tilde{\theta}^*|\widetilde{Z})\tilde{q}(\widetilde{X}, \tilde{\theta}|\widetilde{X}^*, \tilde{\theta}^*, \widetilde{Z})}{p(\widetilde{X}, \tilde{\theta}|\widetilde{Z})\tilde{q}(\widetilde{X}^*, \tilde{\theta}^*|\widetilde{X}, \tilde{\theta}, \widetilde{Z})} \tag{16}$$

where

$$p(\widetilde{X}, \tilde{\theta}|\widetilde{Z}) = \frac{p(\widetilde{X}|\widetilde{Z}, \tilde{\theta})p(\widetilde{Z}|\tilde{\theta})p(\tilde{\theta})}{p(\widetilde{Z})} \tag{17}$$

By (17), it is natural to choose the proposal distribution as

$$\tilde{q}(\widetilde{X}^*, \tilde{\theta}^*|\widetilde{X}, \tilde{\theta}, \widetilde{Z}) = \mathrm{q}_{\tilde{\theta}}(\tilde{\theta}^*|\widetilde{Z})p_{\tilde{\theta}^*}(\widetilde{X}^*|\widetilde{Z}), \tag{18}$$

then (16) becomes

$$\eta = \frac{p_{\tilde{\theta}^*}(\widetilde{Z})p(\tilde{\theta}^*)\mathrm{q}_{\tilde{\theta}^*}(\tilde{\theta}|\widetilde{Z})}{p_{\tilde{\theta}}(\widetilde{Z})p(\tilde{\theta})\mathrm{q}_{\tilde{\theta}}(\tilde{\theta}^*|\widetilde{Z})}. \tag{19}$$

The idea behind this ratio is that algorithm first propose $\tilde{\theta}^*$ from $\mathrm{q}_{\tilde{\theta}}(\cdot|\widetilde{Z})$ in subsection III-C, then calculate the marginal density $p_{\tilde{\theta}}(\widetilde{Z}) = \int p_{\tilde{\theta}}(\widetilde{X}, \widetilde{Z})d\widetilde{X}$. This approach is the called MMH algorithm. In general, computing $\int p_{\tilde{\theta}}(\widetilde{X}, \widetilde{Z})d\widetilde{X}$ is intractable so its approximation, namely the PMMH algorithm, is proposed [19] by using an approximation of $\int p_{\tilde{\theta}}(\widetilde{X}, \widetilde{Z})d\widetilde{X}$ for the MH acceptance ratio where sampling $\widetilde{X}^*$ from $p_{\tilde{\theta}^*}(\widetilde{X}^*|\widetilde{Z})$ is presented in subsection III-A. Convergence of the distribution of samples generated from PMMH is discussed and proved in [19, Theoren 4 and its proof]. Thus the MH acceptance ratio for PMMH is

$$\hat{\eta} = \frac{\hat{p}_{\tilde{\theta}^*}(\widetilde{Z})p(\tilde{\theta}^*)\mathrm{q}_{\tilde{\theta}^*}(\tilde{\theta}|\widetilde{Z})}{\hat{p}_{\tilde{\theta}}(\widetilde{Z})p(\tilde{\theta})\mathrm{q}_{\tilde{\theta}}(\tilde{\theta}^*|\widetilde{Z})}. \tag{20}$$

Choosing an initial $\tilde{\theta}$ for the MC plays an important role for computational cost so we choose an estimate from GM-PHDT to obtain $\tilde{\theta}$ for an initial state of an MC. The Pseudo code for the PMMH-MTT algorithm is given in Algorithm 1, which will be explained next.

Algorithm 1 :PMMH-MTT Algorithm

Input: Given $\widetilde{Z}$, p_{S_t}, p_{D_t}, $\kappa_t, b(\cdot)$ for $t \in \mathcal{T}$ and L the maximum number of iterates in the MC.
Output: S_X is the sequence of $\widetilde{X}$; $S_{\tilde{\theta}}$ the sequence of $\tilde{\theta}$; and $S_{\hat{\tilde{p}}}$ the sequence of $\hat{\tilde{p}}_{\tilde{\theta}}(\widetilde{Z})$ given in (31).
At iteration $l = 1$:

- run GM-PHDT to obtain $\widetilde{X}$; finding $\tilde{\theta}$ from $\widetilde{X}$; denoting $S_X(l) = \widetilde{X}$, $S_{\tilde{\theta}}(l) = \tilde{\theta}$;
- calculate $\tilde{w}_t(\widetilde{X}_{1:t}), t = 1, \ldots, T$ using (23), and $S_{\hat{\tilde{p}}}(l)$ using (31).

At iteration $l > 1$:

- propose $\tilde{\theta}^* \sim \mathrm{q}_{S_{\tilde{\theta}}(l-1)}(\cdot|\widetilde{Z})$ described in subsection III-C;
- run an SMC algorithm 2 in subsection III-A; then sample $\widetilde{X}^* \sim \hat{p}_{\tilde{\theta}^*}(\cdot|\widetilde{Z})$; calculate $\hat{\tilde{p}}_{\tilde{\theta}^*}(\widetilde{Z})$ using (31) and the acceptance rate using (32)

$$\upsilon = \min\{1, \hat{\eta}\} = \min\left\{1, \frac{\hat{\tilde{p}}_{\tilde{\theta}^*}(\widetilde{Z})\mathrm{q}_{\tilde{\theta}^*}(S_{\tilde{\theta}}(l-1)|\widetilde{Z})}{\hat{\tilde{p}}(S_{\tilde{\theta}}(l-1))\mathrm{q}_{S_{\tilde{\theta}}(l-1)}(\tilde{\theta}^*|\widetilde{Z})}\right\};$$

- accept the proposal $\widetilde{X}^*$ and $\tilde{\theta}^*$ if $\upsilon \geq u$ where $u \sim Unif[0,1]$, set $S_X(l) = \widetilde{X}^*$, $S_{\hat{\tilde{p}}}(l) = \hat{\tilde{p}}_{\tilde{\theta}^*}(\widetilde{Z})$ and $S_{\tilde{\theta}}(l) = \tilde{\theta}^*$. Otherwise $S_X(l) = S_X(l-1), S_{\tilde{\theta}}(l) = S_{\tilde{\theta}}(l-1)$, $S_{\hat{\tilde{p}}}(l) = S_{\hat{\tilde{p}}}(l-1)$.

A. Sequential Monte Carlo Algorithm

In SMC algorithms [19, p.272], for any given $\tilde{\theta}_{1:t}$ the posterior densities $\{p_{\tilde{\theta}_{1:t}}(\widetilde{X}_{1:t}|\widetilde{Z}_{1:t}), t \geq 1\}$ are sequentially approximated by the weighted samples $\{\widetilde{X}^n_{1:t}, W^n_t\}_{n=1}^N$

$$p_{\tilde{\theta}_{1:t}}(\widetilde{X}_{1:t}|\widetilde{Z}_{1:t}) \approx \hat{p}_{\tilde{\theta}_{1:t}}(\widetilde{X}_{1:t}|\widetilde{Z}_{1:t}) = \sum_{n=1}^{N} W^n_t \delta(\widetilde{X}^n_{1:t} - \widetilde{X}_{1:t}).$$

where $\delta(\cdot)$ is a Dirac delta and $\sum_{n=1}^N W^n_t = 1$. The idea is to find a sample $\widetilde{X}_1$ conditionally on $\widetilde{Z}_1$ and $\tilde{\theta}_1$, and then to sequentially sample $\widetilde{X}_2, \widetilde{X}_3, \ldots$ to build up a sample $\widetilde{X}_{1:t}$. The pseudo code for SMC is given in Algorithm 2, which will be explained next. SMC first approximates $p_{\tilde{\theta}_1}(\widetilde{X}_1|\widetilde{Z}_1)$ using the proposal density $q_{\tilde{\theta}_1}(\widetilde{X}_1|\widetilde{Z}_1)$ to generate N particles $\widetilde{X}^n_1, n = 1, \ldots, N$ and use the ratio between two densities $p_{\tilde{\theta}_1}(\widetilde{X}_1|\widetilde{Z}_1)p_{\tilde{\theta}_1}(\widetilde{Z}_1)$ and $q_{\tilde{\theta}_1}(\widetilde{X}_1|\widetilde{Z}_1)$ for calculating the weight. The proposal is typically chosen as a Gaussian mixture. At time $t > 1$, we also generate N particles. When generating particle n at time t, we first select which particle at time $t-1$ to propagate according to the weight of the particles. This is done in (26) in Algorithm 2 where $\mathrm{P}(\cdot|W_t)$ is the discrete probability distribution on the set $\{1, \ldots, N\}$ such that $\mathrm{P}(n|W_t) = W^n_t \geq 0$ with $W_t := (W^1_t, \ldots, W^N_t)$. That is, we select to propagate particle A^n_{t-1}, which then becomes the parent of particle n at time t. Then a new multi-target state $\widetilde{X}^n_t$ is sampled using the proposal distribution $q_{\tilde{\theta}_{1:t}}(\cdot|\widetilde{Z}_t, \widetilde{X}^{A^n_{t-1}}_t)$ and the sample $\widetilde{X}^n_{1:t}$ is formed by concatenating $(\widetilde{X}^{A^n_{t-1}}_{1:t}, \widetilde{X}^n_t)$. The proposal distribution at time t is

$$q_{\tilde{\theta}_{1:t}}(\widetilde{X}_{1:t}|\widetilde{Z}_{1:t}) = p_{\tilde{\theta}_{1:t-1}}(\widetilde{X}_{1:t-1}|\widetilde{Z}_{1:t-1})q_{\tilde{\theta}_t}(\widetilde{X}_t|\widetilde{Z}_t, \widetilde{X}_{t-1}) \tag{21}$$

then the corresponding weight is (note that $\tilde{\theta} = \tilde{\theta}_{1:T}$ where $\tilde{\theta}_{t+1:T}$ does not contribute the calculation of $w_t(\widetilde{X}_{1:t})$)

$$\begin{aligned} w_t(\widetilde{X}_{1:t}) &= \frac{p_{\tilde{\theta}}(\widetilde{X}_{1:t}|\widetilde{Z}_{1:t})p(\widetilde{Z}_t)}{q_{\tilde{\theta}}(\widetilde{X}_{1:t}|\widetilde{Z}_{1:t})} \\ &= \frac{p_{\tilde{\theta}}(\widetilde{X}_{1:t-1}|\widetilde{Z}_{1:t-1})g(\widetilde{Z}_t|\widetilde{X}_t, \tilde{\theta}_t)f(\widetilde{X}_t|\widetilde{X}_{t-1}, \tilde{\theta}_t)}{p_{\tilde{\theta}}(\widetilde{X}_{1:t-1}|\widetilde{Z}_{1:t-1})q_{\tilde{\theta}_t}(\widetilde{X}_t|\widetilde{Z}_t, \widetilde{X}_{t-1})} \\ &= \frac{g(\widetilde{Z}_t, \tilde{\theta}_t|\widetilde{X}_t)f(\widetilde{X}_t|\widetilde{X}_{t-1})}{q_{\tilde{\theta}_t}(\widetilde{X}_t|\widetilde{Z}_t, \widetilde{X}_{t-1})}\frac{1}{p(\tilde{\theta}_t)} \end{aligned} \tag{22}$$

where $f(\widetilde{X}_t|\widetilde{X}_{t-1}, \tilde{\theta}_t) = f(\widetilde{X}_t|\widetilde{X}_{t-1})$. Denote

$$\tilde{w}_t(\widetilde{X}_{1:t}) = \frac{g(\widetilde{Z}_t, \tilde{\theta}_t|\widetilde{X}_t)f(\widetilde{X}_t|\widetilde{X}_{t-1})}{q_{\tilde{\theta}_t}(\widetilde{X}_t|\widetilde{Z}_t, \widetilde{X}_{t-1})} \tag{23}$$

where $g(\widetilde{Z}_t, \tilde{\theta}_t|\widetilde{X}_t)$ is given in (12). This is the step given by (24) and (27) in Algorithm 2. The final nth particle can be written as $\widetilde{X}^n_{1:T} = (\widetilde{X}^{B^n_1}_1, \ldots, \widetilde{X}^{B^n_T}_T)$ where $B^n_{1:T}$ is the ancestral lineage with $B^n_T = n$, $B^n_t := A^{B^n_{t+1}}_t$ for $t \in \mathcal{T}\backslash\{T\}$.

Algorithm 2 : **SMC Algorithm**

Input: $\tilde{\theta}$, $\widetilde{Z}$, $q_{\tilde{\theta}}(\cdot|\tilde{Z})$, and number of samples N.
Output: $\{\widetilde{X}^n_{1:t_0}, w_{t_0}(\widetilde{X}^n_{1:t_0}), W^n_{t_0}, A^n_{1:t_0-1}\}_{n=1}^N$, $t_0 \leq T$
At time $t = 1$: for $n = 1, \ldots, N$,
– sample $\widetilde{X}^n_1 \sim q_{\tilde{\theta}_1}(\cdot|\tilde{Z}_1)$,
and use (23) to compute and to normalize the weights

$$\tilde{w}_1(\widetilde{X}^n_1) = \frac{g(\widetilde{Z}_1, \tilde{\theta}_1|\widetilde{X}^n_1) f(\widetilde{X}^n_1)}{q_{\tilde{\theta}_1}(\widetilde{X}^n_1|\widetilde{Z}_1)}, \tag{24}$$

$$W^n_1 = w_1(\widetilde{X}^n_1) / \sum_{m=1}^N w_1(\widetilde{X}^m_1) = \tilde{w}_1(\widetilde{X}^n_1) / \sum_{m=1}^N \tilde{w}_1(\widetilde{X}^m_1), \tag{25}$$

– assign $\boldsymbol{W}_1 := (W^1_1, \ldots, W^N_1)$.
At time $t = 2, \ldots, t_0$: for $n = 1, \ldots, N$,
– sample

$$A^n_{t-1} \sim \mathrm{P}(\cdot|\boldsymbol{W}_{t-1}); \tag{26}$$

– sample $\widetilde{X}^n_t \sim q_{\tilde{\theta}_t}(\cdot|\widetilde{Z}_t, \widetilde{X}^{A^n_{t-1}}_{t-1})$; set $\widetilde{X}^n_{1:t} = (\widetilde{X}^{A^n_{t-1}}_{1:t-1}, \widetilde{X}^n_t)$;
and use (23) to compute and to normalize the weights

$$\tilde{w}_t(\widetilde{X}^n_{1:t}) = \frac{g(\widetilde{Z}_t, \tilde{\theta}_t|\widetilde{X}^n_t) f(\widetilde{X}^n_t|\widetilde{X}^{A^n_{t-1}}_{t-1})}{q_{\tilde{\theta}_t}(\widetilde{X}^n_t|\widetilde{Z}_t, \widetilde{X}^{A^n_{t-1}}_{t-1})}, \tag{27}$$

$$W^n_t = \tilde{w}_t(\widetilde{X}^n_{1:t}) / \sum_{m=1}^N \tilde{w}_t(\widetilde{X}^m_{1:t}), \tag{28}$$

– assign $\boldsymbol{W}_t := (W^1_t, \ldots, W^N_t)$.

Once particles are obtained, an approximation of the target distribution $p_{\tilde{\theta}}(\widetilde{X}|\widetilde{Z})$ is given by

$$\hat{p}_{\tilde{\theta}}(\widetilde{X}|\widetilde{Z}) = \sum_{n=1}^N W^n_T \delta(\widetilde{X} - \widetilde{X}^n). \tag{29}$$

Furthermore, the SMC algorithm provides an estimate

$$\hat{p}_{\tilde{\theta}}(\widetilde{Z}) = \prod_{t=1}^T \frac{1}{N} \sum_{n=1}^N w_t(\widetilde{X}^n_{1:t}) = \frac{\prod_{t=1}^T \frac{1}{N} \sum_{n=1}^N \tilde{w}_t(\widetilde{X}^n_{1:t})}{p(\tilde{\theta})} \tag{30}$$

of the marginal likelihood

$$\int p_{\tilde{\theta}}(\widetilde{Z}, \widetilde{X}) d\widetilde{X} = \prod_{t=1}^T \left(\int w_t(\widetilde{X}_{1:t}) q_{\tilde{\theta}_{1:t}}(\widetilde{X}_t|\widetilde{Z}_t, \widetilde{X}_{t-1}) d\widetilde{X}_t \right)$$

where $q_{\tilde{\theta}_1}(\widetilde{X}_1|\widetilde{Z}_1, \widetilde{X}_0) = q_{\tilde{\theta}_1}(\widetilde{X}_1|\widetilde{Z}_1)$. Denote

$$\hat{\tilde{p}}_{\tilde{\theta}}(\widetilde{Z}) = \prod_{t=1}^T \frac{1}{N} \sum_{n=1}^N \tilde{w}_t(\widetilde{X}^n_{1:t}). \tag{31}$$

Then by (30) and (31), (20) can be written as

$$\hat{\eta} = \frac{\hat{\tilde{p}}_{\tilde{\theta}^*}(\widetilde{Z}) \mathrm{q}_{\tilde{\theta}^*}(\tilde{\theta}|\widetilde{Z})}{\hat{\tilde{p}}_{\tilde{\theta}}(\widetilde{Z}) \mathrm{q}_{\tilde{\theta}}(\tilde{\theta}^*|\widetilde{Z})}. \tag{32}$$

where $p(\tilde{\theta})$ is canceled out in acceptance rate $\hat{\eta}$.

B. Representation of Proposal Distribution

Now we proceed to construct the proposal distribution $\mathrm{q}_{\tilde{\theta}}(\cdot|\widetilde{Z})$ given $\tilde{\theta}$. Designing this proposal distribution involves constructing a MC on the space of $\tilde{\theta}$ which is not easy. Instead we construct an MC on an equivalent space whose element $\tilde{\theta}_\tau$ is defined as *auxiliary track*

$$\tilde{\theta}_\tau = (k, t, j_0, \ldots, j_m) \tag{33}$$

with $k = \boldsymbol{l}(\tau)$, $t = \mathbf{t}_0(\tau)$ and $\tilde{\theta}_{t+i}(k) = (j_i, k)$ for $i = 0, \ldots, m$. Note that if $j_i > 0$, then j_i represents $\theta_{t+i}(k)$ in (13) and is the measurement index at time $t + i$. Hence, the auxiliary track $\tilde{\theta}_\tau$ contains information about the measurements associated with a track τ and inherits the following properties from track τ: 1) label i.e. $\boldsymbol{l}(\tilde{\theta}_\tau) = \boldsymbol{l}(\tau)$, 2) the instances of the track existence i.e. $\mathfrak{T}(\tilde{\theta}_\tau) = \mathfrak{T}(\tau)$, 3) the initial time of appearance $\mathbf{t}_0(\tilde{\theta}_\tau) = \mathbf{t}_0(\tau)$. Thus the track τ and the auxiliary track $\tilde{\theta}_\tau$ are used interchangeably in terms of these properties. Conversely, given $\tilde{\theta}_\tau$, we also find $\tilde{\theta}$ by denoting the measurement index of $\tilde{\theta}_\tau$ at time $t' \in \mathfrak{T}(\tilde{\theta}_\tau)$ as

$$\mathfrak{I}_{t'}(\tilde{\theta}_\tau) = j_{t'-t}.$$

We also know that the target $\boldsymbol{l}(\tau)$ is undetected if $\mathfrak{I}_{t'}(\tilde{\theta}_\tau) = 0$ or generates the measurement $z_{\mathfrak{I}_{t'}(\tilde{\theta}_\tau)}$ if $\mathfrak{I}_{t'}(\tilde{\theta}_\tau) > 0$. Similar to track hypothesis ω, if we denote

$$\tilde{\theta}_\omega = \left\{ \tilde{\theta}_\tau : \tau \in \omega \right\},$$

then $\tilde{\theta}_\omega$ represents ω and is called a *auxiliary track hypothesis*. $\tilde{\theta}_\omega$ and $\tilde{\theta}$ are equivalent representations of the combination between track labels and their measurement indices. Given $\tilde{\theta}_\omega$, then augmented auxiliary function $\tilde{\theta}_t$ (where $t = 1, \ldots, T$) assigns target labels to measurement indices at time t, and it is defined as $\emptyset$ if $t \notin \bigcup_{\tilde{\theta}_\tau \in \tilde{\theta}_\omega} \mathfrak{T}(\tilde{\theta}_\tau)$, otherwise it is defined as

$$\tilde{\theta}_t(\boldsymbol{l}(\tilde{\theta}_\tau)) = (\mathfrak{I}_t(\tilde{\theta}_\tau), \boldsymbol{l}(\tilde{\theta}_\tau)), \qquad \tilde{\theta}_\tau \in \tilde{\theta}_\omega. \tag{34}$$

Thus constructing an MC on the space of $\tilde{\theta}$ is equivalent to constructing an MC on the space of $\tilde{\theta}_\omega$. At time t, we denote the clutter of track hypothesis ω by

$$\boldsymbol{\Lambda}_t(\omega) = \left\{ z_j \in Z_t : j \notin \bigcup_{\tau \in \omega} \{\mathfrak{I}_t(\tilde{\theta}_\tau)\} \right\}. \tag{35}$$

The proposal distribution of $\tilde{\theta}_\omega$ is denoted by $\underline{q}(\tilde{\theta}_{\omega^*}|\widetilde{Z}, \tilde{\theta}_\omega)$. The equivalent proposal distribution on the $\tilde{\theta}$ space is denoted by $\mathrm{q}_{\tilde{\theta}}(\tilde{\theta}^*|\widetilde{Z}) = \underline{q}(\tilde{\theta}_{\omega^*}|\widetilde{Z}, \tilde{\theta}_\omega)$ where $\tilde{\theta}$ and $\tilde{\theta}^*$ correspond to $\tilde{\theta}_\omega$ and $\tilde{\theta}_{\omega^*}$ respectively.

C. Construction of Proposal Distribution

This section presents the construction of the proposal distribution $\underline{q}(\tilde{\theta}_{\omega^*}|\widetilde{Z}, \tilde{\theta}_\omega)$. In order to reduce the computational cost, we first make the following assumptions.

(A.1) The maximum speed of any target is $\bar{v}$.
(A.2) The maximum number of consecutive missed detection for any track is $\bar{d}$, ($\bar{d} \geq 1$).

$\bar{d}$ in Assumption (A.2) can e.g. be chosen such that the probability of $\bar{d}$ consecutive missed detections is below an acceptable threshold (the choice of $\bar{d}$ is discussed in [21]).

The proposal distribution $\underline{q}(\tilde{\theta}_{\omega^*}|\widetilde{Z},\tilde{\theta}_\omega)$ is constructed using $N_{PM}=15$ proposal moves [22] which are classified into twelve groups in Table I where $\mathfrak{m}$ is a parameter to indicate the types of proposal moves. The moves in groups I, II, III, and X are from [21]. If a group consists of two moves, the moves are inverses of each other. If a group contains only one move, the move and its reverse move are the same.

Table I: **List of Proposal Moves**

Group	Type	$\mathfrak{m}$
I	Birth(B)/Death (D)	1/12
II	Split(S)/ Merge(M)	7/5
III	Extension (E)/Reduction (R)	2/8
IV	Extension Merge(EM)/Birth Merge(BM)	4/14
V	Switch (Sw)	6
VI	Backward Extension(BE)/Backward Reduction(BR)	3/9
VII	Extension Merge (EM)/Delete Split (DS)	4/13
VIII	Extension Merge (EM)/Birth Merge (BM)	4/14
IX	Birth Merge (BM)/Birth Merge (BM)	14/14
X	Update (Up)	10
XI	Point Update (PUp)	11
XII	Update-Merge-Extension (UpMerE)	15

We use more proposal moves than in [21] in order to speed up the convergence to the stationary distribution and to reduce the chance that the MC is trapped in certain states for a long time. If we only consider 8 proposal moves as in [21], the probability of Extension merge move is $1/64$ since it can only be created by an extension move followed by a Merge move which have probability $1/8$ each. With 15 moves its probability is $1/15$. The relationships between the N_{PM} proposal moves are illustrated in Figure 1 on the space of $\tilde{\theta}_\omega$ with $\bar{d}=2$ and track gate $m^*=3$ (for details on the moves and their interpretation see [22, p.117]).

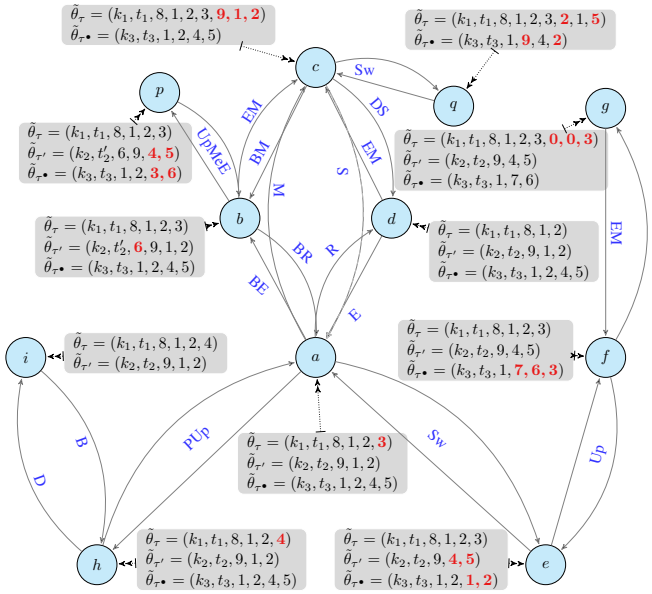


Figure 1: The fifteen moves of the MC on the space of $\tilde{\theta}_\omega$ where $t_3=t_1+3$, $t_2=t_3+1$ and $t_2'=t_2-1$. Each node represents a state of an MC and an arrow describes that the current $\tilde{\theta}_\omega$ moves to a new $\tilde{\theta}_{\omega^*}$.

Now we construct $\underline{q}(\tilde{\theta}_{\omega^*}|\widetilde{Z},\tilde{\theta}_\omega)$. Given $\widetilde{Z},\tilde{\theta}_\omega$, let $P_{\widetilde{Z},\tilde{\theta}_\omega}(\mathfrak{m})$ be a set of all $\tilde{\theta}_{\omega^*}$ which can be constructed from a proposal move type $\mathfrak{m}$, $\mathfrak{m}=1,\ldots,N_{PM}$ that satisfies Assumption (A.1) and (A.2). Specifically,

- If $\tilde{\theta}_\omega=\emptyset$, only a Birth move is proposed i.e. $P_{\widetilde{Z},\tilde{\theta}_\omega}(\mathfrak{m})=\emptyset$ for $\mathfrak{m}\neq 1$.
- If $|\tilde{\theta}_\omega|=1$ i.e. only one track exists, neither Merge, Extension Merge, Update-Merge-Extension moves nor Switch move occur i.e. $P_{\widetilde{Z},\tilde{\theta}_\omega}(\mathfrak{m})=\emptyset$ for $\mathfrak{m}=4,5,6,15$.

Based on this construction, $\tilde{\theta}_{\omega^*}$ is chosen as follows: first the type of move $\mathfrak{m}$ is chosen uniformly at random (u.a.r) from the possible moves, and then $\tilde{\theta}_{\omega^*}$ is chosen u.a.r from $P_{\widetilde{Z},\tilde{\theta}_\omega}(\mathfrak{m})$.

$\tilde{\theta}_{\omega^*}$ specifies a track hypothesis ω^* and $\tilde{\theta}^*$ is the corresponding sequence of augmented auxiliary functions of $\widetilde{\mathbf{X}}_{1:T}(\omega^*)$. Hence whenever $\widetilde{X}$ is drawn from $\mathrm{q}(\cdot|\widetilde{Z},\tilde{\theta}^*)$ in the SMC algorithm 2, there exists a track hypothesis ω^* such that $\widetilde{X}=\widetilde{\mathbf{X}}_{1:T}(\omega^*)$. Knowing whether a measurement is clutter or a potentially target-generated measurement will reduce the computations when sampling from the proposal distribution $\underline{q}(\tilde{\theta}_{\omega^*}|\widetilde{Z},\tilde{\theta}_\omega)$. Next we introduce the set of all measurements that could potentially have been generated from the same target (see [21] for more detail). The introduction of this set will reduce the number of possible auxiliary tracks $\tilde{\theta}_\tau$ associated with the proposal moves introduced below.

Given a measurement z at time t, a measurement z' at time $t+d$, $d=1,2,\ldots,\bar{d}+1$ is called a $d-$*neighbor* of z if $\|z'-z\|\leq d\bar{v}$ where $z'\in Z_{t+d}$. The $d-$*neighborhood* of z at time t, denoted by $L_d(z,t)$, consists of all d-neighbors of z,

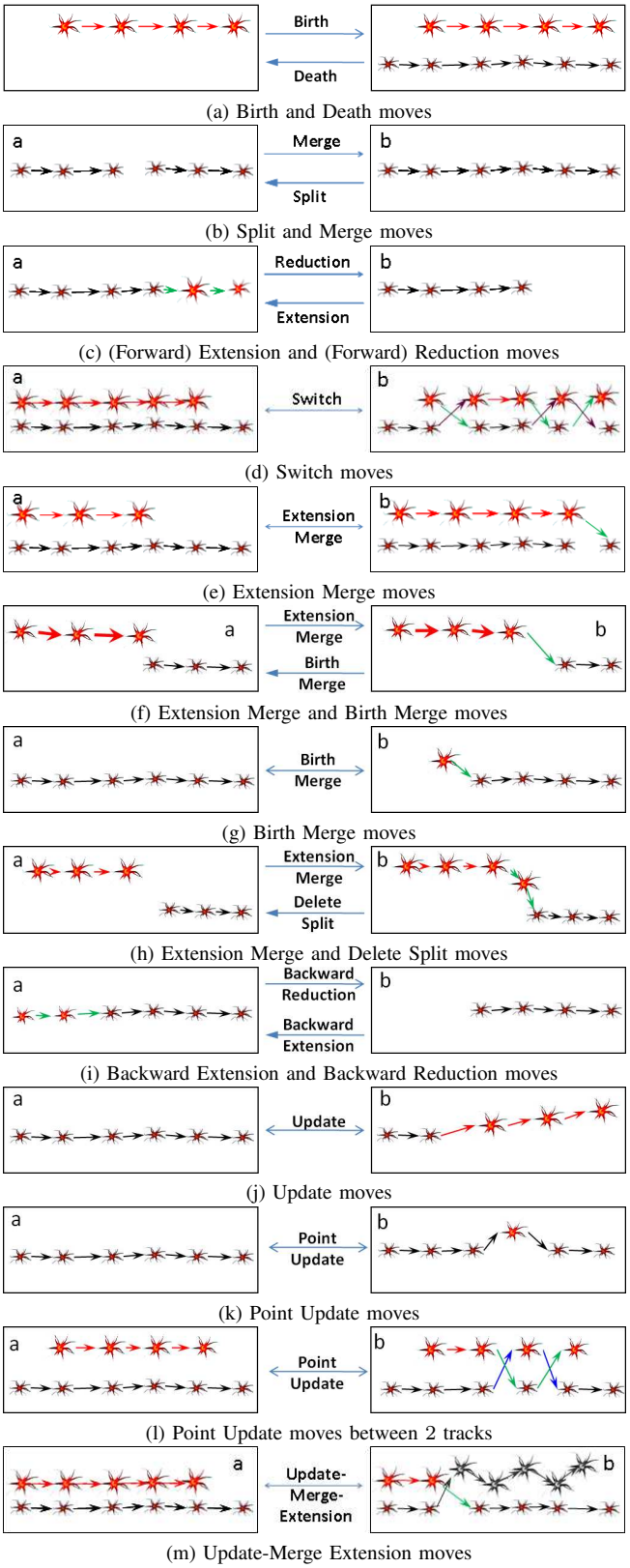
$$L_d(z,t)=\{z'\in Z_{t+d}:\|z'-z\|\leq d\bar{v}\}. \tag{36}$$

The idea is that if a measurement z is generated from a target k at time t then the measurements in $L_d(z,t)$ are the only possible measurements generated by target k at time time $t+d$. The next subsection details how the proposal distribution associated with the N_{PM} proposal moves is constructed using the neighborhoods.

1) Proposal Moves: Figure 2 illustrates the 15 proposal moves. More detail on these moves can be found on [22].

(a) **Birth and death moves (Figure 2a):** A Birth move adds a new track τ^* such that $|\mathfrak{T}(\tau^*)|\geq m^*$ (the duration of existence greater than or equal to m^*) to the track hypothesis ω while keeping all other tracks as before, forming the proposed track hypothesis $\omega^*=\omega\cup\{\tau^*\}$. Its reverse move, a death move is constructed so that it may revert to the initial track after a birth move. A track τ^* is removed from ω while keeping all other tracks as before, forming $\omega^*=\omega\setminus\{\tau^*\}$.

(b) **Split and Merge moves (Figure 2b):** In a Split move a track τ^* with $|\mathfrak{T}(\tau^*)|\geq 2m^*$ (the duration of existence greater than or equal to $2m^*$) is split into two tracks τ and τ' where $\mathbf{t}_0(\tau')=\mathbf{t}_f(\tau)+1$, $\mathbf{t}_f(\tau)$ denotes the last existence time index of track τ. If $\mathbf{t}_0(\tau')>\mathbf{t}_f(\tau)+1$ this move becomes the Delete Split move (see point (e) below). The reverse, a Merge move, can be applied to any two tracks $\tau,\tau'\in\omega$ in which the first target-generated measurement of the target $\boldsymbol{l}(\tau')$ is a $d-$neighbor ($d=1$) of the last target-generated measurement of the target $\boldsymbol{l}(\tau)$. If $d>1$ the move is called an Extension Merge move (see point (e) below).



(a) Birth and Death moves

(b) Split and Merge moves

(c) (Forward) Extension and (Forward) Reduction moves

(d) Switch moves

(e) Extension Merge moves

(f) Extension Merge and Birth Merge moves

(g) Birth Merge moves

(h) Extension Merge and Delete Split moves

(i) Backward Extension and Backward Reduction moves

(j) Update moves

(k) Point Update moves

(l) Point Update moves between 2 tracks

(m) Update-Merge Extension moves

Figure 2: Illustration of 15 proposal moves in groups.

(c) **Extension and Reduction move (Figure 2c):** The objective of the Extension move [21] is to extend the duration of a track by one or more states. In reverse, the Reduction move is applied to remove a track by the last state or the last few states but not below m^*.

(d) **Switch move (Figure 2d):** This move considers the possibility that two targets move so close to each other and cross each other. In Figure 2d, given $\tilde{\theta}_\tau = (k, j_0, \ldots, j_4)$ and $\tilde{\theta}_{\tau'} = (k', j'_0, \ldots, j'_5)$, the Switch move is proposed to form $\tilde{\theta}^*_\tau = (k, j_0, j'_2, j'_3, j_3, j'_5)$ and $\tilde{\theta}^*_{\tau'} = (k', j'_0, j'_1, j_1, j_2, j'_4, j_4)$.

(e) **Extension Merge move/Birth Merge move and Extension Merge move/Delete Split move:** An Extension Merge move is a combination of an Extension move and a Merge move. It may be self-reversible (Figure 2e). A reverse of the Extension Merge move where a new born track merged with a current track is called a Birth Merge move and is a combination of a Birth move and a Merge move (Figure 2f). The Birth Merge move may be self-reversible (Figure 2g). Another reverse of the Extension Merge move is a Delete Split move in which a current track is split into two tracks after deleting some states (Figure 2h).

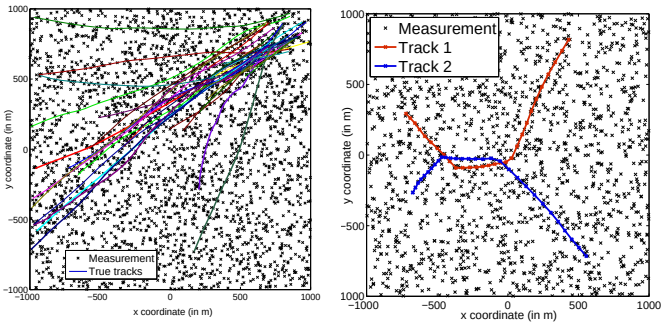
(f) **Backward Extension move and Backward Reduction move (Figure 2i):** A Backward Extension move extends a track backwards in time while a Backward Reduction move removes initial states.

(g) **Update move and Point Update move:** The Update move modifies the state from time t_0 and onwards of the track $\tau \in \omega$ where t_0 is not the first existing time of the target τ (Figure 2j). A Point Update move modifies a single state of a track by modifying the measurement index at that time (see Figures 2k and 2l).

(h) **Update-Merge-Extension move (Figure 2m):** An Update-Merge-Extension move is a combination of an Update, a Merge and an Extension move. This self-reverse move is first proposed as an Update move for the top track τ of Figure 2m.a, then this track is merged to the bottom track τ' of Figure 2m.a. The remaining initial states of track τ' is extended as (Forward) Extension move.

IV. Numerical Results

In this section, we demonstrate the PMMH-MTT on simulated examples. The performances is evaluated using the Optimal Sub-pattern Assignment distance (OSPA) metric [23] and is presented in subsection IV-A. Two examples are used to illustrate the PMMH-MTT. The first example shows that the algorithm is able to handle a large number of closely-spaced and crossing tracks. The second example illustrates the capability to handle situation where two targets cross and move close each other for a long time.



(a) True Tracks of example 1

(b) True Tracks of example 2

Figure 3: Ground true tracks of examples 1 and 2.

A. OSPA metric

Let $X = \{x_1, ..., x_m\}$ and $Y = \{y_1, ..., y_n\}$ be two finite sets and assume that $m < n$. The set X with smaller cardinality is initially chosen as a reference. We want to determine the assignment between the m points of X and the n points of Y that minimizes the sum of distances, subject to the constraint that distances are capped at a preselected maximum or cut-off value c. This minimum sum of distances can be interpreted as the total localization error, which is assigned to the points in Y by giving the points in X as reference. All points which remain unassigned are charged with c the maximum error value. These errors can be interpreted as cardinality errors which are penalized at the maximum rate. The total error committed is then the sum of the localization error and the cardinality error. Remarkably, the per target error obtained by normalizing total error by n (the largest cardinality of the two given sets) is a proper metric.

The OSPA metric $\bar{d}_p^{(c)}$ is defined as follows. Let $\bar{d}^{(c)}(x, y) := \min(c, \|x - y\|)$ for $x, y \in \mathcal{X}$, and Π_k denotes the set of permutations on $\{1, 2, ..., k\}$ for any positive integer k. Then, for $p \geq 1, c > 0$,

- if $m \leq n$:
$$\bar{d}_p^{(c)}(X,Y) = \left[\frac{1}{n}\left(\min_{\pi\in\Pi_n}\sum_{i=1}^{m}\bar{d}^{(c)}(x_i, y_{\pi(i)})^p + c^p(n-m)\right)\right]^{\frac{1}{p}}$$
- if $m > n$: $\bar{d}_p^{(c)}(X,Y) := \bar{d}_p^{(c)}(Y,X)$; and
- if $m = n = 0$: $\bar{d}_p^{(c)}(X,Y) := \bar{d}_p^{(c)}(Y,X) = 0$

The OSPA distance is interpreted as a $p-$th order per-target error, comprised of a $p-$th order per-target localization error and a $p-$th order per-target cardinality error. Precisely, for $p < \infty$ these components are given by

- if $m \leq n$:
$$\bar{e}_{p,\text{loc}}^{(c)}(X,Y) := \left(\frac{1}{n}\min_{\pi\in\Pi_n}\sum_{i=1}^{n}\bar{d}^{(c)}(x_i, y_{\pi(i)})^p\right)^{\frac{1}{p}},$$
$$\bar{e}_{p,\text{card}}^{(c)}(X,Y) := \left(\frac{c^p(n-m)}{n}\right)^{\frac{1}{p}}.$$
- if $m > n$:
$\bar{e}_{p,\text{loc}}^{(c)}(X,Y) = \bar{e}_{p,\text{loc}}^{(c)}(Y,X), \bar{e}_{p,\text{card}}^{(c)}(X,Y) = \bar{e}_{p,\text{card}}^{(c)}(Y,X)$.

The decomposition of the OSPA metric into separate components is usually not necessary for performance evaluation, but may provide valuable additional information.

This metric is suitable for evaluating the MTT problem because at each time it considers not only the error between the number of estimated targets and the number of true targets but also the error between the position of estimated targets and the position of the true targets. The order parameter p determines the sensitivity of the metric to outliers, and the cut-off parameter c determines the relative weighting of the penalties assigned to cardinality and localization errors. In simulation, we choose $c = 80$, $p = 2$.

B. Example 1

1) Ground truths: The surveillance area is the square region $\mathcal{R} = [-1000, 1000] \times [-1000, 1000](m^2, m^2)$ so $V = 4 \times 10^6(m^2)$, and the sampling interval $T_s = 1$ (second). The state vector is $x_t = [\xi_t, \zeta_t, v_{\xi_t}, v_{\zeta_t}]^{Tr}$ where x^{Tr} denotes the transpose and (ξ_t, ζ_t) is the target position in the 2D Cartesian plane and (v_{ξ_t}, v_{ζ_t}) is its velocity $t = 1, \ldots, T = 100$. Linear state and measurement models are used

$$x_t = Ax_{t-1} + v_{t-1}, \qquad z_t = Cx_t + w_t \tag{37}$$

where

$$A = \begin{bmatrix} 1 & 0 & T_s & 0 \\ 0 & 1 & 0 & T_s \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}^{Tr},$$

and where v_t and w_t are zero mean Gaussian process with covariance Q and R, respectively

$$Q = \sigma_v^2 \begin{bmatrix} \frac{T_s^2}{4} I_2 & \frac{T_s}{2} I_2 \\ \frac{T_s}{2} I_2 & I_2 \end{bmatrix}, R = \sigma_w^2 I_2$$

$\sigma_v = 5$ is the standard deviation of the velocity process noise; σ_w is the standard deviation of the measurement noise.

38 targets move from top right or middle of the surveillance area to bottom left, and middle of the surveillance area to top right. Targets appear from $J = 24$ possible locations and can be born at any time in these J possible locations according to a Poisson process with intensity

$$\gamma_t(x) = \sum_{i=1}^{J}\frac{1}{J}\mathcal{N}(x; m_\gamma^{(i)}, P_\gamma) \tag{38}$$

where $P_\gamma = diag(P)$, $P = [100, 100, 25, 25]$ are used to model spontaneous births in the vicinity of $m_\gamma^{(i)}, i = 1, \ldots, J$. Note that $m_\gamma^{(i)}$ specifies both the location and velocity of the targets. The initial speed of the targets as specified by $m_\gamma^{(i)}$ is less than 150 meters per second. Target spawning is not considered in this example so $b(\tilde{x}|\widetilde{X}_{t-1}) = \gamma_t(x)$. The expected number of target $\mu_f(X_{t-1}) = 0.1$. Whenever, a target is born, we form augmented single target state by assigning that target a label different from existing targets during $\mathcal{T}$.

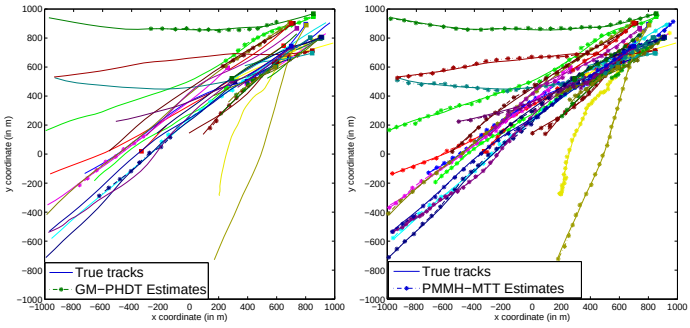
Each target x survives with probability $p_{S_t}(x) = P_S = 0.99$ and is detected with probability $p_{D_t}(x) = P_D = 0.8$. The detected measurements are immersed in clutter modeled as a Poisson RFS Λ_t with intensity

$$\kappa_t = \lambda_c u$$

where $u = 1/V$ is the uniform density over the surveillance region and $\lambda_c = 50$. Thus the average number of clutter returns per unit volume is $\langle \kappa_t, 1 \rangle = \lambda_c$. We tested the performance of the proposed algorithm on this example.

2) PMMH - MTT: The track is confirmed if a target exists at least 3 consecutive times so $m^* = 3$. In the PMMH-MTT, we choose the maximum speed $\bar{v}$ to be $150(m/s)$ and $\bar{d}$ is chosen as 4 such that the probability that a target is detected at least one in 5 scans is $1 - (1 - P_D)^5 = 0.9997$. All other parameters, statistical distribution etc are the same as in the ground truth. With these parameters the average number of 1, 2, 3, 4 and 5 neighbors of each potential track measurement is 62.

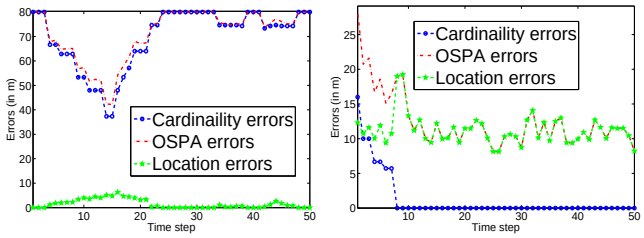
3) Numerical result and discussion: In the absence of a proper estimator we use PMMH-MTT Algorithm 1 and we stop when at least 50 accepted updates are associated with the Point Update move, subject to that the MC has been run at least 100000 times, which indicates that the algorithm has "settled" on a fixed number of targets. In this example, we found successfully 38 trajectories of targets (as illustrated in Figure 3a together with the false measurements). Due to the many closely-spaced targets and large number of false measurements, the filtering technique either performed poorly (such as GM-PHDT in Figure 4a because GM-PHDT use pruning and merging technique to obtain the estimates).



(a) Estimates from filtering technique (b) Estimates from PMMH-MTT

Figure 4: Estimated tracks are plotted against ground truth.

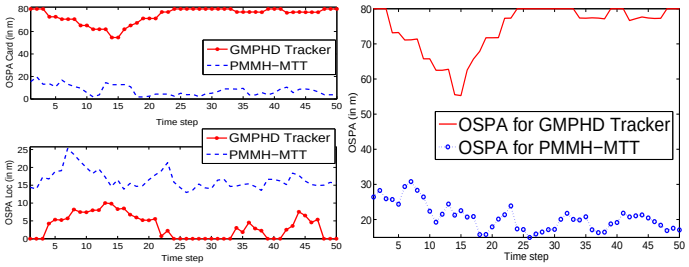
The average of OSPA errors of the last 10000 samples from the PMMH-MTT (Figure 5b) is much lower than the OSPA errors for GM-PHDT (Figure 5a) because the target number is incorrectly estimated using GM-PHDT. Figure 5b shows that the performance of the PMMH-MTT is able to deal successfully with target labels and positions. Indeed, the average OSPA error (Figure 5b) is the same as the average location errors when there is no cardinality errors except during the time between 1 and 7, when a track exists but only is detected at time $1, 3$ and 7. Otherwise, the PMMH-MTT is able to track successfully the number of targets and its states. Note that the location errors for the GM-PHDT is much lower than the PMMH-MTT because there are many targets are not tracked in GM-PHDT.



(a) GM-PHDT (b) PMMH-MTT

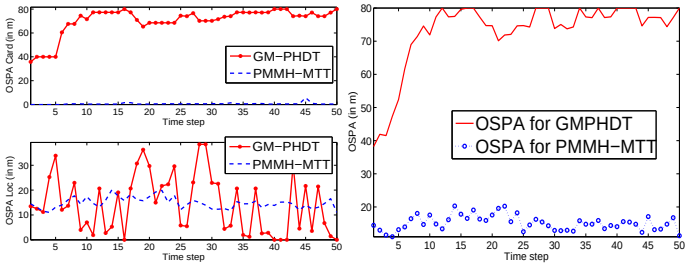
Figure 5: OSPA metric.

To check the average performance we repeated the previous scenario 99 times with the same ground truth (true tracks) but new measurements generated according to the same distribution. We calculated the average OSPA error between ground truths and the last 10000 samples of each Monte Carlo run using OSPA metric. The results in Figure 6 shows that the PMMH-MTT performed well compared to the filtering technique when tracking a large number of closely-spaced and crossing tracks. Note that the location errors for the GM-PHDT is lower than the PMMH-MTT because many targets are not tracked in GM-PHDT (Figure 7a). The PMMH-MTT performed better in term of target number. We repeated the previous scenario with the same ground truth and $P_D = 0.9$, and then the result was plotted in Figure 7. The performance of Figure 6 and 7 showed that the PMMH-MTT is sensitive to the probability of detection. By the nature of MC Monte Carlo (MCMC), there is a trade-off between the performance and the computation time where the state of the MC (the estimate of target trajectories) converges to the target distribution.



(a) Location and cardinality errors. (b) The OSPA errors

Figure 6: Errors of GM-PHDT and PMMH-MTT estimates (10000x100 estimates) using OSPA metric over 100 Monte Carlo runs with $P_D = .8$.



(a) Location and cardinality errors. (b) The OSPA errors

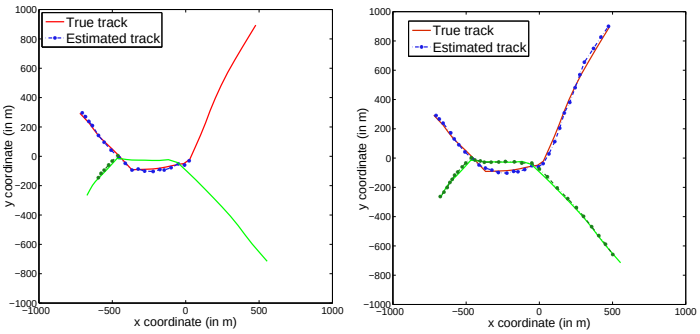
Figure 7: Errors of GM-PHDT and PMMH-MTT estimates (10000x100 estimates) using OSPA metric over 100 Monte Carlo runs with $P_D = .9$.

C. Example 2

1) Ground truths: This example considers the situation where two tracks appear at time 1 and move close to each other from time $t = 10$ to time $t = 17$. The exception are at time $t = 9$ where the target make turns such that they move close together and at time $t = 18$ they separate from each other as shown in Figure 3b. The targets follow the same system model with the same parameter values as in Example 1 with intensity as in (38) but with $J = 2$ and $m^{(1)} = [-700, 300, 20, -30]^{Tr}$, $m^{(2)} = [-700, -250, 20, 30]^{Tr}$ and $T = 30$.

2) PMMH - MTT: The track is confirmed if a target exists at least 5 consecutive times so $m^* = 5$. In the PMMH-MTT, it is assumed that targets appear at any time $t = 1, \ldots, T$ with the same intensity as in subsection IV-C1. The parameters $\bar{v}, \bar{d}$ are chosen as in Example 1. All other parameters, statistical distribution etc are the same as in the ground truth. With these parameters the average number of 1, 2, 3, 4 and 5 neighbors of each potential track measurement is 35.

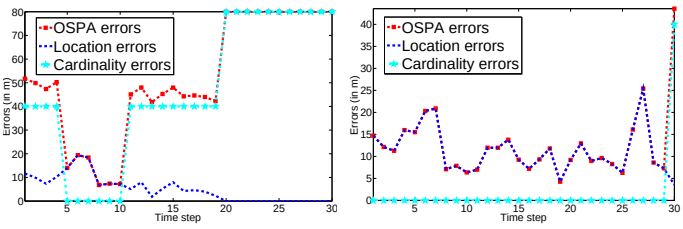
3) Numerical result and discussion: The result in Figure 8b shows that without knowing the number of targets the algorithm handles this scenario successfully by tracking the correct tracks even when the tracks are moving close to each other from time 10 to time 17. The algorithm recovers all of the target states. GM-PHDT on the other hand have difficulties with this scenario due to the unexpected turns at time $t = 9$ and 18 and due to that the targets are undetected at time indices in the vicinities of the time indices 9 an 18.



(a) Initial Estimates and True Tracks (b) Final Estimates and True Tracks

Figure 8: Estimated tracks

The average OSPA errors between state estimates and ground truth from the last 10000 samples are plotted in Figure 9b. In this figure, the performance of the PMMH-MTT improves the initial estimate efficiently by correctly tracking the number of targets at most times (Figure 9b). The location errors for the PMMH-MTT are similar to the OSPA errors because there is no cardinality errors.

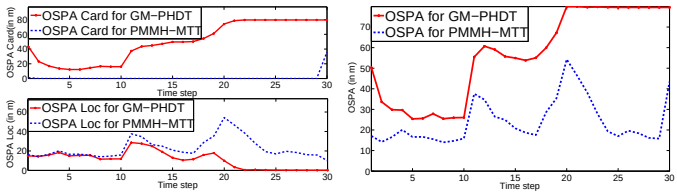


(a) GM-PHDT (b) PMMH-MTT

Figure 9: OSPA metric

In Figure 9b, for the PMMH-MTT the average OSPA errors are the same as the average local errors because there is no average cardinality errors while for the GM-PHDT in Figure 9a, the OSPA errors are much larger than the local errors during the periods from 0 to 4 and from 11 to 30 due to the cardinality errors albeit no or very small location errors.

Similar to Example 1, we repeated the previous scenario 99 times with the same ground truth (true tracks) but new measurements generated according to the same distribution. We also calculated the average OSPA errors between ground truths and the last 10000 samples of each Monte Carlo run. Figure 10b shows that the PMMH-MTT performed better than the GM-PHDT and was able to track the targets when they made unexpected turns and moved close to each other.



(a) Cardinality and local errors. (b) OSPA errors.

Figure 10: average errors of 100 Monte Carlo runs (10000x100 estimates)

V. Conclusion

In this paper, we had proposed a batch MTT technique for solving the problem where a large number of targets move close to each other and cross each other in a dense clutter. The problem was formulated in an RFS framework using Bayesian statistics. An expression for the multi-target distribution was derived and a numerically tractable PMMH-MTT was developed to draw independent samples from it. A key ingredient in the PMMH-MTT was the construction of the proposal distribution based on the proposal moves in section III-C1. A simulation was successfully carried out on a moderately difficult scenario with medium probability of detection ($P_D = 0.8$). The trajectories of a variable number of targets were tracked successfully under two challenging scenarios. Tracking performance was reliable compared to standard filtering methods. However, the run-time cost is high for the batch method 5 hours (example 1) and 2 hours (example 2) on a standard PC while the running time for the GM-PHDT is 30 minutes for example 1 and 20 minutes for example 2 but the results are unreliable. In order for an MC to converge to the stationary distribution (distribution of ground truth) in a reasonable time, the initial state, transition matrix (proposal moves in our case) must be selected. A good estimator which combine samples obtained by Markov chains should also be developed. These are still the issues for further research. In practice, this algorithm is useful when the filtering techniques do not give a satisfactory performance. Particle MCMC methods also attract researchers to track multiple targets such as in [24]. With super computer, the computation cost will be no problem in batch methods for challenging problems which cannot be solved reliably by online methods. However, it is also of interest to explore different methods to reduce the computation.

Acknowledgment

The first author would like to thank Erik Weyer for discussions, proof readings and comments. The work of the second

author is supported by the Australian Research Council under Future Fellowship scheme FT0991854.

REFERENCES

[1] Y. Bar-Shalom and T. Fortmann, *Tracking and data association*. Academic Press, 1988.

[2] S. Blackman, *Multiple-target tracking with radar applications*. Norwood, MA: Artech House, 1986.

[3] J.-R. Larocque, J. Reilly, and W. Ng, "Particle filters for tracking an unknown number of sources," *Signal Processing, IEEE Transactions on*, vol. 50, no. 12, pp. 2926 – 2937, dec 2002.

[4] D. Reid, "An algorithm for tracking multiple targets," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 843 – 854, 1979.

[5] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Multi-target tracking using joint probabilistic data association," in *Proceedings of the 19th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes*, vol. 19, 1980, pp. 807 –812.

[6] D. Musicki and R. Evans, "Joint integrated probabilistic data association: JIPDA," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, no. 3, pp. 1093 – 1099, Jul. 2004.

[7] J. Vermaak, S. Godsill, and P. Perez, "Monte carlo filtering for multi target tracking and data association," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 41, no. 1, pp. 309–332, 2005.

[8] R. Mahler, "The search for tractable Bayesian multitarget filters," in *Proceedings of SPIE - Conference on Signal and Data Processing of Small Targets* , Drummond, OE, Ed., vol. 4048, Apr, 24-27, 2000, pp. 310–320.

[9] ——, "Introduction to multisource-multitarget statistics and its applications," Lockeed Martin Technical monograph, Tech. Rep., 2000.

[10] ——, *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, Inc., 2007.

[11] B.-T. Vo and B.-N. Vo, "Labeled random finite sets and multi-object conjugate priors," *Signal Processing, IEEE Transactions on*, vol. 61, no. 13, pp. 3460–3475, 2013.

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

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

[14] B.-N. Vo and W. Ma, "The Gaussian mixture probability hypothesis density filter," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, p. 4091, 2006.

[15] K. Panta, D. Clark, and B.-N. Vo, "Data association and track management for the Gaussian mixture probability hypothesis density filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 3, pp. 1003–1016, 2009.

[16] O. Erdinc, P. Willett, and Y. Bar-Shalom, "Probability hypothesis density filter for multitarget multisensor tracking," in *In Proceedings of the 8th International Conference on Information Fusion*, vol. 1, July 2005.

[17] R. Mahler, "PHD filters of higher order in target number," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 4, pp. 1523 –1543, Oct. 2007.

[18] B.-T. Vo, B.-N. Vo, and A. Cantoni, "Analytic implementations of the cardinalized probability hypothesis density filter," *IEEE Transactions on Signal Processing*, vol. 55, no. 7, p. 3553, 2007.

[19] C. Andrieu, A. Doucet, and R. Holenstein, "Particle Markov chain Monte Carlo methods," *Jounral of the Royal Statistical Society Series B-Statistical Methodology*, vol. 72, Part 3, pp. 269–342, 2010.

[20] A.-T. Vu, B.-N. Vo, and R. Evans, "Particle Markov Chain Monte Carlo for Bayesian Multi-target Tracking," in *Proceedings of the 14th International Conference on Information Fusion*, Jul.5-8, 2011.

[21] S. Oh, S. Russell, and S. Sastry, "Markov Chain Monte Carlo Data Association for Multi-Target Tracking," *IEEE Transactions on Automatic Control*, vol. 54, no. 3, pp. 481–497, Mar. 2009.

[22] A.-T. Vu, " Particle Markov Chain Monte Carlo for Random Finite Set based Multi-target Tracking," Ph.D. dissertation, Department of Electrical and Electronic Engineering, Univerisity of Melbourne, August 2011.

[23] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A Consistent Metric for Performance Evaluation of Multi-Object Filters," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3447–3457, Aug. 2008.

[24] S. Y. L. Jiang, SS. Singh, " Bayesian tracking and parameter learning for non-linear multiple target tracking models," Univerisity of Cambridge, Tech. Rep., CUED/F-INFENG/TR.692. In preparation.

Anh-Tuyet Vu received the bachelor degree in mathematics from Chi Minh City University of Pedagogy in Vietnam, master degree in Science technology Education from La Trobe University in Australia and Ph.D degree in engineering from the University Melbourne in Australia. Her research interests are theory of random finite sets, Markov chain Monte Carlo methods, estimation and filtering, data fusion, and target tracking.

Ba-Ngu Vo received his Bachelor degrees jointly in Science and Electrical Engineering with first class honors in 1994, and PhD in 1997. He had held various research positions before joining the department of Electrical and Electronic Engineering at the University of Melbourne in 2000. In 2010, he joined the School of Electrical Electronic and Computer Engineering at the University of Western Australia as Winthrop Professor and Chair of Signal Processing. Currently he is Professor and Chair of Signals and Systems in the Department of Electrical and Computer Engineering at Curtin University. Prof. Vo is a recipient of the Australian Research Council's inaugural Future Fellowship and the 2010 Australian Museum Eureka Prize for Outstanding Science in support of Defense or National Security. His research interests are Signal Processing, Systems Theory and Stochastic Geometry with emphasis on target tracking, robotics, computer vision and space situational awareness. He is best known as a pioneer in the random set approach to multi-object filtering.

Rob Evans was born in Melbourne, Australia, in 1947. After completing a BE degree in Electrical Engineering at the University of Melbourne in 1969, he worked as a radar systems engineering officer with the Royal Australian Airforce.

He completed a PhD in 1975 at the University of Newcastle followed by postdoctoral studies at the Laboratory for Information and Decision Systems at MIT, and the Control and Management Department at Cambridge University.

In 1977 he took up an academic position at the University of Newcastle, where he served as Head of the Department of Electrical and Computer Engineering and Co-Director of the ARC Centre on Industrial Control Systems.

In 1992 he moved to the University of Melbourne, where he has served as Head of the Department of Electrical and Electronic Engineering, Research Leader for the Cooperative Centre for Sensor Signal and Information Processing, Director of the Centre for Networked Decision Systems, Dean of Engineering, and Director of NICTA Victoria. He is currently Head of the Department of Electrical and Electronic Engineering and Director of the Defence Science Institute.

His research has ranged across many areas including theory and applications in industrial control, radar systems, signal processing and telecommunications. He is a Fellow of the Australian Academy of Science, a Fellow of the Australian Academy of Technological Sciences and Engineering, and Fellow of IEEE.