UK-PHD Filter:
An Iterated-Update Implementation

SCJ Robertson[*]

2^{nd} December, 2016

Introduction

A Gaussian Mixture (GM) Probability Hypothesis Density (PHD) Filter was implemented as multi-target tracker for EMSS's X-Range project. The filter makes use of a heuristic iterated update for each of the sensors as a more principled approach is currently infeasible. This approach also makes use of the Unscented Transform (UT) to account for non-linearities in the measurement model.

[*] This document's structure is mostly based on M. Fröhle's masters thesis [28], which is useful guide to multi-target filtering with an emphasis on implementation rather than theory.

1. Multi-target Bayes Filtering

The reader is assumed to have some knowledge of standard Bayes filtering, as this section extends the concept using a Random Finite Set (RFS) framework. The exact mathematical details of RFS will neglected here, the focus being more on general concepts.

1.1 Representing Multiple Target Systems

When tracking multiple targets for an extended period of time, the number of targets will obviously vary. At some time, t, the number of existing targets can be collected into a set:

$$X_t = \{\boldsymbol{x}_t^1, \boldsymbol{x}_t^2, \ldots, \boldsymbol{x}_t^N\} \subset E_S \tag{1.1}$$

Here $\boldsymbol{x}_t^i$ is some random vector representing a target's current state. E_S is the state space, introduced here purely for convenience.

The received measurements at time t are now also collected into a set:

$$Z_t = \{\boldsymbol{z}_t^1, \boldsymbol{z}_t^2, \ldots, \boldsymbol{z}_t^M\} \subset E_O \tag{1.2}$$

Here E_O is the measurement space, which needn't be distinct from E_S. Not every element of Z_t is necessarily generated by an existing target, just as a target need not generate a measurement. False and missed detections are naturally inherent to the problem.

1.2 Random Finite Set Framework

The size of the sets in Equations 1.1 and 1.2 vary randomly at each time step as targets are born and die and measurements are missed and clutter falsely detected. It is therefore natural to wish for a model which describes not just the uncertainty of an individual target's state but also the total number of current targets and measurements. Random Finite Set (RFS) Theory provides the necessary framework, but the mathematics is non-trivial, see [36, 38, 39][1]. Both

[1] Perhaps start with [36].

the cardinality of a set as well as its elements must be treated as Random Variables (RVs). Therefore, discrete distributions describing the cardinality of the sets are required as well as a families of continuous joint distributions describing the states of each random vector $\boldsymbol{x}_t^i$ and $\boldsymbol{z}_t^j$ [28, 31].

1.3 Multi-target Bayes Filter

The general multi-target Bayes filter can be derived just as easily as a standard Bayes filter, merely substituting random vectors for random sets. While it is excessive, it will be done just to introduce a shared terminology and notation.

1.3.1 A Single Sensor Bayes Filter

It will first be assumed that the system as a single sensor which collects multiple measurements detected at time t into a single set. Before observing the measurements, the system's evolution can be represented as a joint distribution over all state and measurement RFS:

$$p(X_0, \dots, X_t, Z_1, \dots, Z_t) = p(X_{0:t}, Z_{1:t}) \tag{1.3}$$

The state RFS, $X_{0:t-1}$, can naturally be inferred from the measurement RFS:

$$\begin{aligned}
p(X_{0:t}|Z_{1:t}) &= \frac{p(X_{0:t}, Z_{1:t})}{p(Z_{1:t})} \\
&= \frac{p(Z_t|X_{0:t}, Z_{1:t-1})p(X_{0:t}, Z_{1:t-1})}{p(Z_t|Z_{1:t-1})p(Z_{1:t-1})} \\
&= \frac{p(Z_t|X_t)p(X_t|X_{0:t-1}, Z_{1:t-1})p(X_{0:t-1}, Z_{1:t-1})}{p(Z_t|Z_{1:t-1})p(Z_{1:t-1})} \\
&= \frac{p(Z_t|X_t)p(X_t|X_{t-1}, Z_{1:t-1})p(X_{0:t-1}, Z_{1:t-1})}{p(Z_t|Z_{1:t-1})p(Z_{1:t-1})} \\
&= \frac{p(Z_t|X_t)p(X_t|X_{t-1}, Z_{1:t-1})p(X_{0:t-1}|Z_{1:t-1})\cancel{p(Z_{1:t-1})}}{p(Z_t|Z_{1:t-1})\cancel{p(Z_{1:t-1})}} \\
&= \eta \; p(Z_t|X_t)p(X_t|X_{t-1}, Z_{1:t-1})p(X_{0:t-1}|Z_{1:t-1})
\end{aligned} \tag{1.4}$$

It is important to restate that the standard Bayes Filter assumptions still hold for RFS:

1. $p(Z_t|X_{0:t-1}) = p(Z_t|X_t)$ - Measurements generated at time t are a only ever a direct result of events occurring at time t.

2. $p(X_t|X_{0:t-1}, Z_{1:t-1}) = p(X_t|X_{t-1}, Z_{1:t-1})$ - The Markov assumption, if the current state is fully specified, or *complete*, then the future is independent from the past. If at time t the full history leading up to the current events is known, then at time $t+1$ there is no need to look further into to the past than t.

The current state RFS, X_t, can be obtained by marginalising out all previous state RFS from Equation 1.4:

$$\begin{aligned}
p(X_t|Z_{1:t}) &= \int p(X_{0:t}|Z_{1:t})\mu_s(dX_{0:t-1}) \\
&= \eta \; p(Z_t|X_t) \int p(X_t|X_{t-1}, Z_{1:t-1}) \left[\int p(X_{0:t-1}|Z_{1:t-1})\mu_s\left(dX_{0:t-2}\right) \right] \mu_s(dX_{t-1})
\end{aligned} \tag{1.5}$$

Here μ_s is some appropriate measure[2], as given in [31, 36, 39]. Equation 1.5 can viewed, with some caution, as a simple update function:

$$\eta \qquad \underbrace{p(Z_t|X_t)}_{\text{Measurement Update}} \qquad \underbrace{\int p(X_t|X_{t-1}, Z_{1:t-1}) \overbrace{\left[\int p(X_{0:t-1}|Z_{1:t-1})\mu_s\left(dX_{0:t-2}\right) \right]}^{\text{Recursion}} \mu_s(dX_{t-1})}_{\text{Prediction}}$$

The current notation is bulky and unnecessarily confusing, adopting a more wieldy notation:

$$\begin{aligned} p_t(X_t|Z_{1:t}) &= \eta \; g_t(Z_t|X_t) \int f_{t|t-1}(X_t|X_{t-1}) p_{t-1}(X_t|Z_{1:t-1}) \mu_s(dX_{t-1}) \\ p_t(X_t|Z_{1:t}) &\propto g_t(Z_t|X_t) p_{t|t-1}(X_{t-1}|Z_{1:t-1}) \end{aligned} \tag{1.6}$$

Where,

- $\int p(X_{0:t-1}|Z_{1:t-1})\mu_s\left(dX_{0:t-2}\right) = p_{t-1}(X_{t-1}|Z_{1:t-1})$ - The propagated *posterior*, $p(X_{t-1}|Z_{1:t-1})$, of the previous state.
- $f_{t|t-1}(X_t|X_{t-1}) = p(X_t|X_{t-1}, Z_{1:t-1})$ - The *Markov* distribution. This distribution over X_t's predicted state is created by pushing the previous state's posterior through the motion model.
- $p_{t|t-1}(X_t|Z_{1:t-1}) = \int f_{t|t-1}(X_t|X_{t-1}) p_{t-1}(X_t|Z_{1:t-1}) \mu_s(dX_{t-1})$ - The predicted *prior* distribution over X_t. $f_{t|t-1}(X_t|X_{t-1}) p_{t-1}(X_t|Z_{1:t-1})$ is a joint distribution over X_t and X_{t-1}, X_{t-1} is then marginalised out.
- $g_t(Z_t|X_t) = p(Z_t|X_t)$ - The *likelihood* distribution. The predicted state of X_t is pushed through the measurement model, creating a distribution over Z_t conditioned on X_t before the measurements are observed.
- $p_t(X_t|Z_{1:t}) = p(X_t|Z_{1:t})$ - The *posterior* distribution, the result of the update function. It represents the complete current belief over X_t.
- The normalizing constant η is to be neglected for convenience.

It should be noted that Equation 1.6 is computationally intractable under most conditions (See Section 1.6) due to multiple integrals over arbitrarily complex distributions [10, 11, 31]. However, the filter has been directly applied to a small amount of targets using partical methods [31]. This directly analogous to the standard Bayes Filter, which only has exact, closed form solutions under linear Gaussian assumptions.

1.3.2 "Mutli-sensor Bayes" Filters

The RFS framework does account for multisensor models, however the approach is currently computationally intractable [22, 23]. Therefore it is necessary to apply heuristic techniques to account for multiple measurement updates. Standard approaches are Iterated-Update (IU) and the more correct Bayes Parallel Combination (BPC), both present potential issues [23].

It is now assumed that there are K-many independent sensors, with sensor j producing a set of measurements at time t:

$$Z_t^j = \left\{ \boldsymbol{z}_t^j, \ldots, \boldsymbol{z}_t^{M(j)} \right\} \tag{1.7}$$

[2] Gross oversimplification: A measure provides a method of measuring the "length" of a set, generalising the idea of measuring interval length on $\mathbb{R}$.

Iterated-Updated

As the name suggests, the measurement update is run sequentially for every set of measurements provided by a sensor. Equation 1.6 is usually modified to:

$$p_t(X_t|Z_{1:t}) \approx \left[\prod_{i=1}^{K} g_t^i(Z_t^i|X_t)\right] p_{t|t-1}(X_t|Z_{1:t-1}) \tag{1.8}$$

While is notation is convenient, it is misleading as it does not properly represent the sequential nature of the updates. A proper description would be more algorithmic, after each update the posterior effectively acts as a prior to the next update. The posterior is used to create a new measurement distribution for a sensor, the new measurements are observed and the old posterior updated.

The quality of the results are dependent on the order of the update [22, 23], noisy sensor measurements must be multiplied in first to ensure more precise sensor measurements are not "wasted". If the prior is conjugate and a mixture of exponential family members then the successive multiplications will lead to an exponential explosion of mixture components.

Bayes Parallel Combination

As mentioned IU is sequential and sensitive to sensor ordering. BPC offers a parallelizable, order invariant alternative to IU, represented follows [23]:

$$p_t(X_t|Z_{1:t}) \approx p_{t|t-1}(X_t|Z_{1:t-1})^{1-K} \left[\prod_{i=1}^{K} g_t^i(Z_t^i|X_t) p_{t|t-1}(X_t|Z_{1:t-1})\right] \tag{1.9}$$

The posterior is first updated in parallel with information from each individual sensor, after which they are combined and any repeated information effectively removed by division.

BPC is more accurate than IU, but its tractability is strongly dependent on the form of the prior. For example, if prior is an mixture of exponential family members the division may be computationally intractable. If the prior is conjugate and a mixture of exponential family members then the combination step will still involve an exponential explosion of mixture components.

1.4 Finite Set Statistics

Finite Set Statistics (FISST) is a differential and integral calculus developed to describe the statistics of RFS [37, 39]. The approach is unified, probabilistic foundation for multisource-multitarget systems, but can be boiled down to relatively simple "turn-the-crank" set of operations [9–11, 37]. These techniques have an almost parallel-worlds correspondence with standard random vector statistical techniques [9, 28, 30, 37].

Applying FISST is also beyond the scope of this document, but comprehensive tutorials can be found in [9, 33, 37, 38]. An *Engineering Introduction* is provided by [38] and [9, 30, 33] provide extensive examples of application.

1.5 The Data Association Problem

RFS and FISST stands in contrast to traditional multi-target filtering techniques, such as Multiple Hypothesis Tracking (MHT), as it is association free. The original filters derived from RFS (See Section 1.6) do not form tracks, their output is an unlabelled set of state vectors with no explicit association with the previous state's output.

Proponents of RFS and FISST are critical of traditional data association techniques [32–34], faulting the treatment of hypotheses as ordinary state variables. Letting $\Omega(Z)$ denote the state of all hypotheses defined for the measurement set Z, association based techniques would compute posterior for a hypothesis ω as:

$$p(\omega|Z) = \frac{p(Z|\omega)p(\omega)}{p(Z)} \tag{1.10}$$

However, each hypothesis ω is an element $\Omega(Z)$ itself depend on Z, therefore put explicitly:

$$p(\omega(Z)|Z) = \frac{p(Z|\omega(Z))p(\omega(Z))}{p(Z)} \tag{1.11}$$

From this it not immediately clear whether $p(\omega(Z))$ is a valid prior as it depends on future information [32]. It possible that the likelihood $p(Z|\omega(Z))$ is not a valid conditional probability as $\omega(Z)$ is itself conditioned on Z [32]. From formal definitions of Bayes Rule, the product of the likelihood and prior must form a joint distribution [32, 33]. It is not clear whether $p(Z|\omega(Z))p(\omega(Z))$ is a valid joint distribution as it may not have a legal marginal in $\omega(Z)$:

$$p(\omega(Z)) = \int p(Z|\omega(Z))p(\omega(Z))dZ \tag{1.12}$$

Following similar reasoning, the normalization constant may not well defined either. It has been suggested that data association may not fit into the Bayesian paradigm [32–34].

1.6 A Review of Existing Filters

This section is a brief review of the filters developed from RFS and FISST, a more in-depth review can be found at [24] and MATLAB implementations at [3]:

1. PHD Filter - Propagates the only first moment of the multi-target posterior with a Poisson RFS [8]. The filter has a closed form solution when operating under linear Gaussian assumptions [13].

2. Cardinalised PHD (CPHD) Filter - The PHD filter underestimated the predicted number of targets under certain conditions, the cardinality distribution of the posterior is now propagated alongside its first moment to account for the error [14]. This also has closed form solutions when operating under linear Gaussian assumptions [14].

3. Multi-target-Multi-Bernoulli (MeMBer) Filters - The filter is able to approximately propagate the whole posterior when assuming a Multi-Bernoulli RFS [15].

4. Cardinalised MeMBer Filters - The cardinality distribution of the posterior is also propagated to account for the same underestimation problems the PHD filter encountered [15].

5. Generalised Labelled Multi-Bernoulli (GLMB) Filter - An exact closed form solution of the multi-target Bayes Filter [16, 25]. It provides provides labelled tracks as an output. It is notably the first ever computationally tractable, provably Bayes optimal multi-target tracker [25].

The GLMB filter is first RFS filter to have integrated tracking, however the filter's implementation is far more complex than previous efforts [16]. Track formation for the PHD and MeMBer filters were usually heuristic, relying on traditional data association techniques [17, 31] (despite their supposed conceptual difficulties).

Research on accurate, tractable multi-sensor models is currently lacking [11, 24]. Most practical multi-sensor implementations make use of an iterated-update [18, 19, 21], with some minor alterations to account for sensor ordering [22, 23].

2. Approximate Multi-target Bayes Filters

In single target filtering, propagating the full posterior is only possible if certain limiting assumptions are made on the system dynamics [47, 48]. In general, the distribution must be approximated and its N^{th} order moments are propagated in its place with the remaining moments assumed to be constant. If the prior is Gaussian, the system linear and only the first two moments of the posterior propagated then the approximation is exact. This operation is, of course, the Kalman Filter.

The multi-target Bayes Filter is no different, only having exact closed form solution under certain assumptions [16, 25]. The PHD filter is an approximate solution, the posterior is approximated by its first order moment, its mean, when propagated forward:

$$p_t(X_t|Z_{1:t}) \approx p_t(X_t|D_t) \tag{2.1}$$

The full posterior can only be recovered from its mean when the SNR is low and the targets move independently from one another [8, 28, 31], when the higher order moments are nearly constant. The PHD filter is very much the Alpha-Beta or Constant Gain Kalman Filter of the RFS world.

2.1 Probability Hypothesis Density

The Probability Hypothesis Density (PHD), or intensity, specifically refers to the to the first order statical moment of a RFS [11, 28, 31]:

$$D_\Xi(\boldsymbol{x}) = \mathbb{E}\left\{\delta_\Xi(\boldsymbol{x})\right\} \tag{2.2}$$

Here $\mathbb{E}$ denotes expectation and

$$\delta_\Xi(\boldsymbol{x}) = \sum_{x \in \Xi} \delta_{\boldsymbol{x}} \tag{2.3}$$

$\delta_\Xi(\boldsymbol{x})$ is the density of a RFS Ξ, represented by Dirac delta functions centred around the random vectors $\boldsymbol{x} \in \Xi$ [28]. Perhaps not obvious from its form, the PHD is a continuous surface in E_S and its peaks give the estimates of the random vector elements of Ξ. Integrating over the surveillance region $S \subseteq E_S$ returns the expected number of targets in S [8, 13, 17, 28]:

$$\overline{N}_\Xi = \mathbb{E}\left\{|\Xi \cap S|\right\} = \int_S D_\Xi(\boldsymbol{x}) d\boldsymbol{x} \tag{2.4}$$

The PHD is not a Probability Density Function (PDF), however it does represent probability mass[1]. It is analogous, and in Section 2.3 exactly the same, to an unnormalized mixture of weighted PDFs. Integrating such a mixture over a region will provide an estimate of the number of components present, their weights' representing their contribution of probability mass to the mixture.

[1] Belief mass would perhaps be more correct [36, 38].

2.2 The PHD-Filter

As mentioned in the section introduction, the PHD Filter approximates a multi-target Bayes filter by propagating only its first order moment, or PHD. The filter is association free and does not explicitly offer track identity. The current state estimates are not presented as a neat set, but must be extracted from the PHD surface using some heuristic method [8, 28].

The PHD recursion involves multiple integrals over arbitrary distributions, the filters are usually implemented by particle methods [8, 13, 28]. This approach also has the added advantage of not imposing any strong limitations on the target dynamics and measurement models. Particle methods, however, are beyond the scope of this document.

A full derivation of the the PHD filter using FISST can be found in [33]. An intuitive physical space derivation, which does not use RFS, is available in [12].

The State set

The current RFS state estimate, X_t, is assumed Poisson and is defined as follows [28, 31]:

$$X_t = \left(\bigcup_{\boldsymbol{x} \in X_{t-1}} S_{t|t-1}(\boldsymbol{x}) \right) \bigcup \left(\bigcup_{\boldsymbol{x} \in X_{t-1}} \Gamma_{t|t-1}(\boldsymbol{x}) \right) \bigcup B_t \tag{2.5}$$

$S_{t|t-1}$ is a RFS of all surviving targets from previous state RFS X_{t-1}. Each existing target is assumed to survive with some state dependent probability p_S and evolve through the Markov distribution $f_{t|t-1}(\cdot|X_{t-1})$. $\Gamma_{t|t-1}$ is the RFS of new targets spawned from previous state's targets; such as a fighter jet launching a missile. B_t is the RFS of all new targets spontaneously birthed at the current time step.

The Measurement Set

The RFS set of measurements, Z_t, is defined as [28, 31]:

$$Z_t = \left(\bigcup_{\boldsymbol{x} \in X_t} \theta_t(\boldsymbol{x}) \right) \bigcup K_t \tag{2.6}$$

θ_t is the RFS of all measurements generated by current targets in the RFS X_t. K_t is an Poisson distributed RFS containing false alarm detections generated by the clutter, its PHD is given by:

$$\kappa_t(\boldsymbol{z}) = \lambda_t u_t(\boldsymbol{z}) \tag{2.7}$$

$u_t(\boldsymbol{z})$ is a uniform distribution over the surveillance region S. λ_t is the average amount of clutter detected per unit "volume" of S, for simplicity, it usually assumed to be a constant.

2.2.1 Prediction

The predicted prior PHD, $D_{t|t-1}(\boldsymbol{x}_t|Z_{1:t-1})$, is defined as follows [8, 13, 17, 28]:

$$D_{t|t-1}(\boldsymbol{x}_t|Z_{1:t-1}) = b_t(\boldsymbol{x}_t) + \int \phi_{t|t-1}(\boldsymbol{x}_t, x_{t-1}) D_{t-1|t-1}(\boldsymbol{x}_{t-1}|Z_{1:t-1}) d\boldsymbol{x}_{t-1} \tag{2.8}$$

Where,

$$\phi_{t|t-1}(\boldsymbol{x}_t, x_{t-1}) = p_S(\boldsymbol{x}_{t-1}) f_{t|t-1}(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) + \gamma_{t|t-1}(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) \tag{2.9}$$

Here $b_t(\boldsymbol{x}_t)$ is the PHD for the spontaneous birth at time t. $p_S(\boldsymbol{x}_t)$ is some state dependent survival probability from time $t-1$ to t. $f_{t|t-1}(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ is a single target Markov distribution and $\gamma_{t|t-1}(\boldsymbol{x}_t, \boldsymbol{x}_{t-1})$ is the PHD of spawned targets.

2.2.2 Measurement Update

The updated posterior PHD, $D_{t|t}(\boldsymbol{x}_t|Z_{1:t})$, is defined as follows [8, 17, 28]:

$$D_t(\boldsymbol{x}_t|Z_{1:t}) = \left[(1 - p_D(\boldsymbol{x}_t)) + \sum_{\boldsymbol{z}\in Z_t} \frac{p_d(\boldsymbol{x}_t)g(\boldsymbol{z}|\boldsymbol{x}_t)}{\kappa_t(\boldsymbol{z}) + \int p_D(\boldsymbol{x}_t)D_{t|t-1}(\boldsymbol{x}_t|Z_{1:t-1})g_t(\boldsymbol{z}|\boldsymbol{x}_t)d\boldsymbol{x}_t}\right] D_{t|t-1}(\boldsymbol{x}_t|Z_{1:t-1}) \tag{2.10}$$

Here $p_D(\boldsymbol{x}_t)$ is the state dependent detection probability. $g(\boldsymbol{z}|\boldsymbol{x}_t)$ is a single target likelihood function. $\kappa_t(\boldsymbol{z}_t)$ is the established clutter intensity.

The posterior is a sum of parts, a scaled predicted prior and an updated, weighted "posterior" for each individual measurement. This scaled prior accounts for the possibility that all genuine measurements go undetected. The prediction stage is exact, the measurement update is approximate and results in a loss of information [8, 30, 32] due to errors in the Poisson approximations.

As an aside, traditional approaches to measurement updates are:

1. *Track-to-Measurement* - An established target generates a measurement, which either goes undetected or is in Z_t. This approach cannot naturally model clutter, as a genuine target cannot generate clutter.

2. *Measurement-to-Track* - A measurement is either generated by clutter or it is caused by an established target. This approach cannot naturally model missed detections, as the measurement is established as detected.

It should noted that the PHD filter is exactly neither of these and attempts to account for both clutter and missed detections. If the clutter intensity, $\kappa_t(\boldsymbol{z})$, for a measurement $\boldsymbol{z}$ is immensely high, the likelihood of its updated posterior naturally decreases.

2.2.3 State Extraction

The updated posterior PHD is continuous surface with peaks indicating the targets' states. The total number are targets and their estimated states must be extracted from the PHD surface. The suggested approach is [8, 13, 17, 28]:

1. Determined the number of targets, $\overline{N}_t$, by integrating the PHD surface over the surveillance region S, as in Equation 2.4.

2. Find the $\overline{N}_t$ greatest peaks on the PHD surface using some peak finding algorithm.

3. Approximate the $\overline{N}_t$-many targets' states using the coordinates of these peaks.

However, such an approach only provides a point estimate of a target's state. To account for variance in a target's state a Gaussian Mixture (GM) can be fitted to PHD surface [12–14], at great computational cost.

2.2.4 Approximate Multi-Sensor Models

As mentioned is Section 1.3.2, the true multi-sensor model must be approximated by IU and BPC. In the literature, IU dominates the GM-PHD implementations [18–21] as there is no computationally tractable method of dividing GMs[2]. Therefore, IU will be presented as a standard method for all multi-sensor PHD filters.

[2] It might be possible to evaluate both mixtures over the same representative region, say S, perform point-to-point division and fit a GM to the resulting surface.

An order invariant IU multi-sensor PHD filter, the product multi-sensor PHD (PM-PHD), has been derived [22, 23]:

$$D_t(\boldsymbol{x}_t) = \overline{N}_t \frac{\left[\prod_{i=1}^{K} g_t^i(Z_t^i|X_t)\right] S_{t|t-1}(\boldsymbol{x}_t)}{\int p_{D,t}(\boldsymbol{x}_t) \left[\prod_{i=1}^{K} g_t^i(Z_t^i|X_t)\right] S_{t|t-1}(\boldsymbol{x}_t) d\boldsymbol{x}_t} \tag{2.11}$$

Where,

$$S_{t|t-1}(\boldsymbol{x}_t) = \frac{1}{\overline{N}_{t-1}} D_{t|t-1}(\boldsymbol{x}_t) \tag{2.12}$$

The PM-PHD is equivalent to the IU update of Section 1.3.2, assuming K-many independent sensors, save for some additional normalization terms. Here $\overline{N}_{t-1}$ is the expected predicted number of targets and $\overline{N}_t$ the expected number of targets after measurement update, as determined by Equation 2.4.

The PM-PHD has a higher computational overhead than the IU-PHD, but is significantly more accurate when the sensors have a low probability of detection [22, 23].

2.3 The Gaussian Mixture-PHD Filter

There is no general closed form solution to PHD filter recursion given in Equations 2.8 and 2.10. However, the recursion is closed under linear Gaussian assumptions [13, 14]. Under these conditions the prior is conjugate, the PHD surface is approximated by a Gaussian Mixture (GM), with target dynamics essentially acting as a loose collection of independent Kalman Filters [13,14,28]. The GM-PHD filter can easily be extended to mildly non-linear systems by merely extending the Kalman filter.

The GM-PHD filter will not be derived, only the main results and assumptions will be presented. A derivation can be found in [13, 14]. The GM-PHD filter is a closed form solution to an approximation of a multi-target posterior under restrictive conditions. It is a weaker approximation of a true multi-target posterior than the standard PHD, but offers an efficient and flexible implementation.

2.3.1 Linear-Gaussian Multi-Target Model

This section will introduce the assumptions necessary to derive the the GM-PHD Filter. The first, and most important, is that the system is Linear-Gaussian, namely:

1. The motion model is a linear function, with added Gaussian noise.

$$\boldsymbol{x}_t = A_t \boldsymbol{x}_{t-1} + B_t \boldsymbol{u}_t + \boldsymbol{\epsilon}_t \tag{2.13}$$

   A_t and B_t are $n \times n$ and $m \times n$ matrices respectively, where n is the dimension of the state vector. $\boldsymbol{\epsilon}_t$ is zero mean Gaussian noise with a covariance matrix R_t.

2. The measurement model is also a linear function, with added Gaussian noise.

$$\boldsymbol{z}_t = C_t \boldsymbol{x}_t + \boldsymbol{\delta}_t \tag{2.14}$$

   C_t is $k \times n$, $\boldsymbol{z}_t$ being k dimensional. $\boldsymbol{\delta}_t$ is zero mean Gaussian noise with a covariance matrix Q_t.

3. The prior distribution must be a GM:

$$b_0(\boldsymbol{x}_0) = \sum_{i=0}^{J_0} w_0^i N(\boldsymbol{x}_0|\boldsymbol{m}_0^i, P_0^i) \tag{2.15}$$

The notation was chosen to be consistent with [48] as well as [13,28]. The proceeding sections require finicky notations; subscripts and superscripts will be dropped whenever the context is obvious.

The PHD of spontaneous births, b_t, must now also be modelled by a GM [13,14]:

$$b_t(\boldsymbol{x}_t) = \sum_{i=1}^{J_{b,t}} w^i_{b,t} \mathcal{N}(\boldsymbol{x}_t | \boldsymbol{m}^i_{b,t}, P^i_{b,t}) \tag{2.16}$$

Similarly, the PHD of spawned targets, $\gamma_{t|t-1}$, is also assumed to GM operating under linear Gaussian conditions [13,14]:

$$\gamma_{t|t-1}(\boldsymbol{\zeta}_t) = \sum_{i=1}^{J_{\gamma,t}} w^i_{\gamma,t} \mathcal{N}(\boldsymbol{\zeta}_t | \boldsymbol{m}^i_{\gamma,t}, P^i_{\gamma,t}) \tag{2.17}$$

The spawned targets are each assumed to obey their own linear motion models:

$$\boldsymbol{\zeta}_t = F^i_{\gamma,t} \boldsymbol{\zeta}_{t-1} + \boldsymbol{d}^i_{\gamma,t-1} \tag{2.18}$$

The spawned target's is modelled to be within the proximity of its parent [13, 14]. Here it presented as an affine transformation of its parents expected state, it forwards it parent's state through some motion model, $F^i_{\gamma,t}$, and then translates it an appropriate distance, $\boldsymbol{d}^i_{\gamma,t-1}$, away.

As an aside, the spawned target model does imply that the PHD-filter must be capable of managing multiple motion models for different types of targets without an explicit classification scheme. It achieves this in a similar manner to a manoeuvring target Kalman Filter, each target is simply propagated through every motion model and all predicted states are added to the mixture [33].

There is no requirement for these Gaussian mixtures to be normalized, their weights should reflect a belief of whether a target is actually present in a given region. If a single Gaussian PDF is added to a any region, the expected number of targets in this region is at least 1, which is fairly strong belief of target presence.

Both the detection and survival probabilities can be generalized to exponential mixtures:

$$p_{S,t}(\boldsymbol{x}_t) = p_{S,t}(\boldsymbol{x}_t) + \sum_{i=1}^{J_{S,t}} w^i_{S,t} \mathcal{N}(\boldsymbol{x}_t | \boldsymbol{m}^i_{S,t}, P^i_{S,t}) \tag{2.19}$$

$$p_{D,t}(\boldsymbol{x}_t) = p_{D,t}(\boldsymbol{x}_t) + \sum_{i=1}^{J_{S,t}} w^i_{S,t} \mathcal{N}(\boldsymbol{x}_t | \boldsymbol{m}^i_{D,t}, P^i_{D,t}) \tag{2.20}$$

However, it should be noted that this representation usually accounts for regions where the target is expected to have low probabilities of survival and detection. Therefore, the mixture weights and covariances may be negative [13, 14]. Great care must be taken to ensure that the weights and covariances of the PHD stay non-negative after merging and pruning (See Section 2.3.4). For the required purposes, $p_{S,t}(\boldsymbol{x}_t)$ and $p_{S,t}(\boldsymbol{x}_t)$ will not represented as mixtures, but rather in their standard state dependent form.

From this point onwards, given the context, the terms GM, PHD and intensity all refer to the PHD-surface and are interchangeable.

2.3.2 Prediction

It is assumed that the previous state's posterior is a GM of the form:

$$D_{t-1}(\boldsymbol{x}_{t-1}) = \sum_{i=1}^{J_{t-1}} w^i_{t-1} \mathcal{N}(\boldsymbol{x}_{t-1} | \boldsymbol{m}^i_{t-1}, P^i_{t-1}) \tag{2.21}$$

Then the predicted PHD is also a GM, given by:

$$D_{t|t-1}(\boldsymbol{x}_t) = \left[D_{S,t|t-1}(\boldsymbol{x}_t) + D_{\gamma,t|t-1}(\boldsymbol{x}_t)\right] + b_t(\boldsymbol{x}_t) \tag{2.22}$$

Where $D_{S,t|t-1}$ is the PHD of surviving targets advanced through the motion model of Equation 2.13:

$$D_{S,t|t-1}(\boldsymbol{x}_t) = p_{S,t}(\boldsymbol{x}_t) \sum_{i=1}^{J_{t-1}} w_{t-1}^i \mathcal{N}(\boldsymbol{x}_t | \boldsymbol{m}_{S,t|t-1}^i, P_{S,t|t-1}^i) \tag{2.23}$$

$$\boldsymbol{m}_{S,t|t-1}^i = A_t \boldsymbol{m}_{t-1}^i \tag{2.24}$$

$$P_{S,t|t-1}^i = A_t P_{t-1}^i A_t^T + R_t \tag{2.25}$$

$D_{\gamma,t|t-1}$ is the PHD of all targets spawned from the previous state's targets as given by Equations 2.17 and 2.18:

$$D_{\gamma,t|t-1}(\boldsymbol{x}_t) = \sum_{i=1}^{J_{t-1}} \sum_{j=1}^{J_{\gamma,t}} w_{t-1}^i w_{\gamma,t}^j \mathcal{N}(\boldsymbol{x}_t | \boldsymbol{m}_{\gamma,t|t-1}^{i,j}, P_{\gamma,t|t-1}^{i,j}) \tag{2.26}$$

$$\boldsymbol{m}_{\gamma,t|t-1}^{i,j} = F_{\gamma,t}^j \boldsymbol{m}_{t-1}^j + \boldsymbol{d}_{\gamma,t-1}^j \tag{2.27}$$

$$P_{\gamma,t|t-1}^{i,j} = F_{\gamma,t}^j P_{t-1}^i \left(F_{\gamma,t}^j\right)^T + R_{\gamma,t-1}^j \tag{2.28}$$

$b_t(\boldsymbol{x}_t)$ is the GM of all targets spontaneously born at time t as given by Equation 2.16. Both $D_{\gamma,t|t-1}(\boldsymbol{x}_t)$ and $b_t(\boldsymbol{x}_t)$ simply raise the intensity of the predicted PHD in pre-defined regions according to the evolution of the system.

If $p_{S,t}$ is not state dependent, the expected number of predicted targets represented by the current PHD can be determined as follows [13, 28]:

$$\overline{N}_{t|t-1} = \overline{N}_{t-1} \left(p_{S,t} + \sum_{i=1}^{J_{\gamma,t}} w_{\gamma,t}^i \right) + \sum_{i=1}^{J_{b,t}} w_{b,t}^i \tag{2.29}$$

Where $\overline{N}_{t-1}$ is the expected number of targets in the previous states posterior. This is still equivalent to Equation 2.4, however all Gaussian mixture components will integrate to 1, therefore the expected number of targets can simply be determined by a discrete sum of the mixture components' weights.

2.3.3 Measurement Update

Supposing the assumptions of the previous sections hold, then the predicted PHD is given by the following GM:

$$D_{t|t-1}(\boldsymbol{x}_t) = \sum_{i=1}^{J_{t|t-1}} w_{t|t-1}^i \mathcal{N}(\boldsymbol{x}_t | \boldsymbol{m}_{t|t-1}^i, P_{t|t-1}^i) \tag{2.30}$$

The posterior intensity at time t can be now be given by the following GM:

$$D_t(\boldsymbol{x}_t) = (1 - p_{D,t}(\boldsymbol{x}_t))\, D_{t|t-1}(\boldsymbol{x}_t) + \sum_{\boldsymbol{z} \in Z_t} D_{D,t}(\boldsymbol{z}, \boldsymbol{x}_t) \tag{2.31}$$

Where,

$$D_{D,t}(\boldsymbol{z}, \boldsymbol{x}_t) = \sum_{i=1}^{J_{t|t-1}} w_t^i(\boldsymbol{z}) \mathcal{N}(\boldsymbol{x}_t | \boldsymbol{m}_{t|t}^i(\boldsymbol{z}), P_{t|t}^i(\boldsymbol{z})) \tag{2.32}$$

$$w_k^i(\boldsymbol{z}) = \frac{p_{D,t}(\boldsymbol{x}) w_{t|t}^i g_t^i(\boldsymbol{z})}{\kappa_t(\boldsymbol{z}) + p_{D,t}(\boldsymbol{x}) \sum_{j=1}^{J_{t|t-1}} w_{t|t-1}^j q_t^j(\boldsymbol{z})} \tag{2.33}$$

$$g_t^i(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z} | C_t \boldsymbol{m}_{t|t-1}^i, C_t P_{t|t-1}^i C_t^T + Q_t) \tag{2.34}$$

$$\boldsymbol{m}_{t|t-1}^i(\boldsymbol{z}) = \boldsymbol{m}_{t|t-1}^i + K_t^i(\boldsymbol{z} - C_t \boldsymbol{m}_{t|t-1}^i) \tag{2.35}$$

$$P_{t|t}^i = \left[I - K_t^i C_t\right] P_{t|t-1}^i \tag{2.36}$$

$$K_t^i = P_{t|t-1}^i C_t^T \left(C_t P_{t|t-1}^i C_t^T + R_t\right)^{-1} \tag{2.37}$$

Equations 2.34 to 2.37 are standard Kalman Filter action for a measurement update [47, 48]. The GM-PHD Filter operates in a very similar manner to the Gaussian Sum Filter [13, 14], as a loose collection of independent Kalman Filters represented by a GM, but the PHD represents all the target states over a physical surveillance region.

If $p_{D,t}$ is not state dependent, the expected number of targets after the measurement update can is given by [13, 28]:

$$\overline{N}_t = \overline{N}_{t|t-1} (1 - p_{D,k}) + \sum_{\boldsymbol{z} \in Z_t} \sum_{j=1}^{J_{t|t-1}} w_t^j(\boldsymbol{z}) \tag{2.38}$$

Here $\overline{N}_{t|t-1}$ is the predicted number of targets given in Equation 2.29.

2.3.4 Pruning and Merging

Propagating an unconstrained GM will lead to an exponential explosion of mixture components at each time step. A simple pruning method can be employed to stem the expansion, discarding all components which have weak weights [13, 14, 28].

The GM-PHD contains a survey of all current targets updated with every available measurement, much like MHT. Many of these components will be close enough together that they can be merged and accurately approximated as a single Gaussian [13, 28]. More nuanced approaches do not merge closely spaced targets with significant weights, rather allowing them to represent distinct targets [19].

2.3.5 State Extraction

After merging each mixture component represents a local maximum, therefore state extraction is relatively straightforward. The height of a mixture component is dependent on its weight and covariance, selecting the $\overline{N}_t$ highest peaks may correspond to targets with weak weights [13, 14]. This is undesirable because the expected number of targets due these peaks is small, even though their peaks are high. To better represent the actual probability mass it is more proper to select the components with larger weights. Targets are now selected as mixture components with weights larger than some predefined threshold, typically taken as 0.5 [13, 18–21].

3. Product-Multisensor Unscented-Kalman PHD Filter

The purpose of the filter is to accurately estimate multiple golf-trajectories in 3D-space given K-many RADAR sensors. The sensors provide only partial state information in the form of Range-Doppler measurements. The IU procedure is roughly equivalent to multilateration, requiring at least 4 sensors to provide an accurate estimate of a target's state.

Naturally; the measurement model is non-linear, but the motion model is to assumed a linear constant velocity model. The measurement update will therefore be approximated using the Unscented-Transform (UT), while the prediction step will be implemented directly as described in Section 2.3.2.

As discussed in Section 1.5, this filter is association free. The filter acts as input to a secondary filter, the Joint Probabilistic Data Association (JPDA) filter, which will perform track association on the processed data. Such an approach was typical before the development of labelled RFS [17, 31].

This section will present the specific algorithms used in the implementation of the PM-UK-PHD filter. For brevity's sake, this filter will be just be referred to as the UK-PHD filter. The algorithms presented here are largely the single-sensor algorithms developed in [13, 28] extended to the PM-PHD of Section 2.2.4.

3.1 The Unscented Transform

The Kalman Filter is a closed form solution to the Bayes Filter under linear-Gaussian conditions. Its efficiency and flexible implementation far outweigh its restrictive assumptions and it since been extended to account for mildly non-linear conditions [48]. All such extensions enforce the same condition; the posterior must be Gaussian.

The UT deterministically samples a set of $2n+1$ representative points, called Sigma Points, from the prior distribution's covariance matrix and passes them through some arbitrary function $\boldsymbol{f}(\cdot)$ [5–7]. The transformed points are then viewed as events generated from some underlying distribution, which is approximated as a Gaussian by its first two moments.

The motivation behind the UT can be found in [5–7] and its application to the Kalman Filter in [4, 6, 48]. An exhaustive derivation of the UKF can also be found in [1].

For the purposes of this document the UT will be viewed as a function which when invoked on the Gaussian distribution:

$$\{\boldsymbol{y}_t, S_t, \Sigma_{\boldsymbol{x},\boldsymbol{y}}\} = \mathrm{UT}_{\boldsymbol{f}}\left(\boldsymbol{x}_t, \Sigma_t, Q_t\right) \tag{3.1}$$

Returns,

1. $\boldsymbol{y}_t$ - The mean of transformed distribution.
2. S_t - The covariance of the transformed distribution.
3. $\Sigma_{\boldsymbol{x},\boldsymbol{y}}$ - The cross-covariance between $\boldsymbol{x}_t$ and $\boldsymbol{y}_t$.

Here Q_t is the covariance matrix of some perturbation noise used to ensure the transformed distribution is non-singular. There is no guarantee that the Sigma points will retain their independence once passed through $\boldsymbol{f}(\cdot)$.

3.2 Prediction

The Prediction step presented in Algorithm 1 is relatively straightforward. As spawning is not applicable, it will be ignored. A weighted Gaussian PDF will be represented as the triple $\{w, \boldsymbol{m}, P\}$, and a GM-PHD will be represented as the collection of triples $\{w_i^i, \boldsymbol{m}_t^i, P_t^i\}_{i=1}^{J_t}$.

The current GM-PHD is predicted by:

- Propagating every surviving target with a legal state through the motion model.
- Adding in all components from the birth PHD into predicted GM-PHD.

Algorithm 1 UK-PHD Prediction.

```
$D_{t-1} \leftarrow \{w_{t-1}^i, \boldsymbol{m}_{t-1}^i, P_{t-1}^i\}_{i=1}^{J_{t-1}}$ Previous state's PHD
$b_t \leftarrow \{w_{b,t}^i, \boldsymbol{m}_{b,t}^i, P_{b,t}^i\}_{i=1}^{J_{b,t}}$ Current state's birth PHD
$\overline{N}_{t-1} \leftarrow$ Expected number of targets in $D_{t-1}$

function PREDICT-STATE($D_{t-1}$, $b_t$, $\overline{N}_{t-1}$)
    $i = 0$                                              ▷ Global component counter
    for $j = 1, \dots, J_{b,t}$ do                       ▷ Add in components from birth PHD
        $i = i + 1$
        $w_{t|t-1}^i = w_{b,t}^j$
        $\boldsymbol{m}_{t|t-1}^i = \boldsymbol{m}_{b,t}^j$
        $P_{t|t-1}^i = P_{b,t}^j$

    for $j = 1, \dots, J_{t-1}$ do                       ▷ Predict surviving targets' state
        if IS_LEGAL_STATE ($\boldsymbol{m}_{t-1}^j$) then    ▷ Remove all targets with illegal states
            $i = i + 1$
            $w_{t|t-1}^i = p_{S,t} w_{t-1}^j$
            $\boldsymbol{m}_{t|t-1}^i = A_t \boldsymbol{m}_{t-1}^j + B\boldsymbol{u}_t$
            $P_{t|t-1}^i = A_t P_{t-1}^j A_t^T + R_t$

    $\overline{N}_{t|t-1} = p_S \overline{N}_{t-1} + \sum_{j=0}^{J_{b,t}} w_{b,t}^j$    ▷ The expected number of predicted targets
    $J_{t|t-1} = i$                                      ▷ Total amount of GM components

    for $j = 1, \dots, J_{t|t-1}$ do                     ▷ Normalisation required by PM-PHD
        $w_{t|t-1}^j = w_{t|t-1}^j / \overline{N}_{t|t-1}$

    $D_{t|t-1} = \{w_{t|t-1}^i, \boldsymbol{m}_{t|t-1}^i, P_{t|t-1}^i\}_{i=1}^{J_{t|t-1}}$    ▷ The predicted PHD

    return $D_{t|t-1}$, $\overline{N}_{t|t-1}$
```

- Calculating the expected number of targets in the GM-PHD.
- Normalising the predicted GM-PHD, using the expected number targets, according to Equation 2.12. This is used to reduce the effects of sensor ordering during the iterated measurement update.

In Algorithm 1, Is_Legal_State is any function which determines whether a target is of interest. As example, in the Eagle Canyon scenario targets travelling predominately along the y-axis probably do not originate from the driving range and can be ignored. This is not a probabilistically sound approach, however such hard-coded rules do offer a large computational savings and provide comparatively accurate practical results.

3.3 Measurement Update

The measurement update step presented in Algorithm 2 is the workhorse of the UK-PHD and its presentation is slightly daunting. Given the predicted prior, $D_{t|t-1}$, and a class of measurement sets from sensor:

$$Z_t = \{Z_t^1, \ldots, Z_t^K\} \tag{3.2}$$

Sequential measurement updates are performed using measurement set, Z_t^i, from sensor i. It is assumed that each sensor i has a unique measurement model based on its position, that when implemented with the UT, is represented as:

$$\left\{\boldsymbol{\eta}_{t|t-1}^j, S_t^j, \Sigma_t^j\right\} = \mathrm{UT}_i\left(\boldsymbol{m}_{t|t-1}^j, P_{t|t-1}^j, Q_t\right) \tag{3.3}$$

Q_t is assumed to be the same for all sensors. Also while presented as a constant, the detection probability, $p_{D,t}$, should be unique to each sensor.

During each iteration of the measurement update:

- The J_t-many prior components are first scaled by the missed detection probability and then added to the posterior GM-PHD.
- Update components, the transformed mean, covariance, Kalman gain and innovation are created for each of the prior's components.
- For each measurement $\boldsymbol{z} \in Z_t^i$, the measurement is introduced as evidence into every update component. Each measurement contributes J_t-many updated components to the posterior mixture.
- All the updated components are now weighted according to Equation 2.33. $\kappa_t(\boldsymbol{z}) = \lambda_t$, the average amount of clutter expected over the entire sensor volume (See Section 2.2).
- The updated components are added to the posterior mixture, the posterior mixture is then pruned using the Merge-Prune algorithm (Algorithm 3).
- The expected number of targets in the pruned posterior GM-PHD is calculated.
- The pruned posterior mixture now serves as the prior for the next iteration.

After the measurement updates are complete, the final posterior is normalised according to Equation 2.11 to reduce the effects of sensor ordering. The expected number of targets, $\overline{N}_t$, was also rounded off after the IU. This is not suggested anywhere in the literature, but it was found to improve the stability of the result over time.

Algorithm 2 UK-PHD Measurement Update.

```
$D_{t|t-1} \leftarrow \{w^i_{t|t-1}, \boldsymbol{m}^i_{t|t-1}, P^i_{t|t-1}\}_{i=1}^{J_{t|t-1}}$ The current state's predicted PHD
$Z_t \leftarrow \{Z_t^1, \dots, Z_t^K\}$ Sensor measurement superset
$\kappa_t \leftarrow$ The clutter intensity
$\overline{N}_{t|t-1} \leftarrow$ Expected number of targets in $D_{t|t-1}$.
$T \leftarrow$ Truncation threshold
$U \leftarrow$ Merging threshold
$J_{\max} \leftarrow$ Maximum number of mixture components

function MEASUREMENT-UPDATE($D_{t|t-1}$, $Z_t$, $\kappa_t$, $T$, $U$, $J_{\max}$)
    $\overline{N}_t = \overline{N}_{t-1}$
    $J_t = J_{t|t-1}$

    for $k = 1, \dots, K$ do                    ▷ A measurement update for every sensor
        $n = 0$                                  ▷ Local component counter

        for $j = 1, \dots, J_t$ do               ▷ Account for missed detections
            $w^j_t = (1 - p_{D,t}) w^j_{t|t-1}$
            $\boldsymbol{m}^j_t = \boldsymbol{m}^j_{t|t-1}$
            $P^j_t = P^j_{t|t-1}$

        for $j = 1, \dots, J_t$ do               ▷ Construct update components
            $\left\{\boldsymbol{\eta}^j_{t|t-1}, S^j_t, \Sigma^j_t\right\} = \mathrm{UT}_k\left(\boldsymbol{m}^j_{t|t-1}, P^j_{t|t-1}, Q_t\right)$
            $K^j_t = \Sigma^j_t (S^j_t)^{-1}$    ▷ Kalman Gain
            $P^j_t = P^j_{t|t-1} - K^j_t S^j_t (K^j_t)^T$    ▷ Innovation

        for $\boldsymbol{z} \in Z^k_t$ do
            $n = n + 1$
            for $j = 1, \dots, J_t$ do           ▷ Introduce evidence into every update component
                $w^{nJ_t+j}_t = p_{D,t}\ w^j_{t|t}\ \mathcal{N}(\boldsymbol{z}|\boldsymbol{\eta}^j_{t|t}, S^j_t)$
                $\boldsymbol{m}^{nJ_t+j}_t = \boldsymbol{m}^j_{t|t} + K^j_t\left(\boldsymbol{z} - \boldsymbol{\eta}^j_{t|t}\right)$
                $P^{nJ_t+j}_t = P^j_t$

            $\Upsilon_{\boldsymbol{z}} = \kappa_t(\boldsymbol{z}) + \sum_{i=1}^{J_t} w^{nJ_t+i}_t$    ▷ Determine likelihood weighting
            for $i = 1, \dots, J_t$ do
                $w^{nJ_t+i}_t = w^{nJ_t+i}_t / \Upsilon_{\boldsymbol{z}}$    ▷ Weight each updated component

        $D_t = \{w^i_t, \boldsymbol{m}^i_t, P^i_t\}_{i=1}^{(n+1)J_t}$    ▷ Posterior PHD
        $D_t =$ MERGE-PRUNE $(D_t, T, U, J_{\max})$    ▷ Prune the PHD components
        $J_t = |D_t|$                            ▷ The number of components in the pruned PHD

        $\overline{N}_t = (1 - p_{D,t})\overline{N}_t + \sum_{i=1}^{J_{t|t}} w^i_t$    ▷ Expected number of targets in $D_t$

    $\Upsilon = \sum_{i=1}^{J_{t|t}} w^i_t$
    for $i = 1, \dots, J_{t|t}$ do
        $w^i_t = \overline{N}_t w^i_t / \Upsilon$    ▷ Normalization for PM-PHD

    return $D_t$, $\overline{N}_t$
```

Algorithm 3 UK-PHD Pruning and Merging.

```
$D_t \leftarrow \{w_t^i, \boldsymbol{m}_t^i, P_t^i\}_{i=1}^{J_t}$ A GM-PHD
$T \leftarrow$ Truncation threshold
$U \leftarrow$ Merging threshold
$J_{\max} \leftarrow$ Maximum number of mixture components

function MERGE-PRUNE($D_t$, $T$, $U$, $J_{\max}$)
    $I = \{i = 1, \dots, J_t | J_t > w_t^i\}$        ▷ Index set for components with significant weights
    $l = 0$        ▷ Pruned component counter

    while $I \neq \{\varnothing\}$ do
        $l = l + 1$
        $j = \arg\max_{i \in I} w_t^i$
        $L = \{i \in I | (\boldsymbol{m}_t^i - \boldsymbol{m}_t^j)^T (P_t^i)^{-1} (\boldsymbol{m}_t^i - \boldsymbol{m}_t^j) \leq U\}$        ▷ All comp. indexes $i$ close to $j$

        $\overline{w}_t^l = \sum_{i \in L} w_t^i$
        $\overline{\boldsymbol{m}}_t^l = \frac{1}{\overline{w}_t^l} \sum_{i \in L} w_k^i \boldsymbol{m}_t^i$
        $\overline{P}_t^l = \frac{1}{\overline{w}_t^l} \sum_{i \in L} w_t^i \left(P_t^i + (\boldsymbol{m}_t^l - \boldsymbol{m}_t^i)(\boldsymbol{m}_t^l - \boldsymbol{m}_t^i)^T\right)$

        $I = I \setminus L$        ▷ Remove all merged components from $I$

    $\overline{D}_t = \{\overline{w}_t^i, \overline{\boldsymbol{m}}_t^i, \overline{P}_t^i\}_{i=1}^{l}$        ▷ The pruned GM-PHD

    if $l > J_{\max}$ then
        $\overline{D}_t = $ SORT $(D_t)$        ▷ Sort according to $\overline{w}_t^i$
        $\overline{D}_t = \{\overline{w}_t^i, \overline{\boldsymbol{m}}_t^i, \overline{P}_t^i\}_{i=1}^{J_{\max}}$        ▷ Choose the $J_{\max}$ largest components

    return $\overline{D}_t$
```

3.4 Pruning and Merging

Algorithm 3 is identical to that of [13, 14]. It first discards all mixture components with weak weights, then it merges together all remaining, closely spaced components into a single Gaussian. If this process does not reduce the below the allowable maximum, then only the remaining $J_{\max}$ largest components are kept.

3.5 State Extraction

Algorithm 4 UK-PHD State Extraction.

```
$D_t \leftarrow \{w_t^i, \boldsymbol{m}_t^i, P_t^i\}_{i=1}^{J_t}$ A GM-PHD
$E_t \leftarrow$ The Extraction threshold

function STATE-EXTRACTION($D_t$, $E_T$)
    $\overline{X}_t = [\varnothing]$                                ▷ The estimated targets' states
    for $i = 1, \dots, J_t$ do
        if $w_k^i \geq E_T$ then
            for $j = 1, \dots,$ ROUND $(w_k^i)$ do
                $\overline{X}_t = [X_t, \boldsymbol{m}_t^i]$
    return $\overline{X}_t$
```

Algorithm 4 is again identical that those given in [13, 14, 28]. No reason is given to the intentional overcounting of components with large weights, it may be a desire to better represent the underlying probability mass.

4. Performance Measures

In single target filtering, any consistent distance metric can be used to measure the distance between the estimated state, $\boldsymbol{x}_t$, and the ground truth $\boldsymbol{y}_t$. In the RFS approach to multi-target tracking, the true target state is given by $X_t = \{\boldsymbol{x}_t^1, \dots, \boldsymbol{x}_t^M\}$ and the ground truth by $Y_t = \{\boldsymbol{y}_t^1, \dots, \boldsymbol{y}_t^N\}$, the sets are unordered[1] and could have differing cardinalities. A consistent performance metric must be capable of handling the complexities of association, by determining an estimated state $\boldsymbol{x}_t^i$'s corresponding state $\boldsymbol{y}_t^j$ in Y, if it exists at all.

4.1 Inconsistent Distance Metrics

Earlier attempts to establish multi-target distance metrics fell short as they were not mathematically consistent and physically intuitive, but they provided the foundation for subsequent devlopments. The original Hausdorff distance does not consider cardinality errors and heavily penalizes outliers arising from clutter [26, 28]. The Optimal Mass Transfer (OMAT) distance attempted to improve on this, but it is incompatible with RFS theory and is undefined for empty sets [27, 28, 32]. These metrics have been included as performance measures in this implementation, despite their shortcomings.

[1] The sets are unordered by definition and relative to one another.

4.2 Optimal Subpattern Assignment

The Optimal Subpattern Assignment (OSPA) distance is based on the earlier OMAT metric [26–28, 32], and is defined as:

$$\overline{d}_p^{(c)} = \left[\frac{1}{n} \min_{\pi \in \Pi_n} \sum_{i=1}^{m} d^{(c)}(\boldsymbol{x}_i, \boldsymbol{y}_{\pi(i)})^p + c^p(n-m) \right]^{\frac{1}{p}} \tag{4.1}$$

Here $\overline{d}_p^{(c)}$ is the distance between the two finite subsets $X = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_m\}$ and $Y = \{\boldsymbol{y}_1, \dots, \boldsymbol{y}_n\}$ of a finite set W. Π_n is the set of all permutations of $\{1, 2, \dots, n\}$. The term $d^{(c)}(\boldsymbol{x}, \boldsymbol{y}) = \min(c, d(\boldsymbol{x}, \boldsymbol{y}))$, where $d(\boldsymbol{x}, \boldsymbol{y})$ is some arbitrary distance. c is a maximum cut-off distance which affects the cardinality errors and defines a maximum search radius for $\boldsymbol{x}_i \in X$ to look for $\boldsymbol{y}_j \in Y$ [27, 28]. The parameter p controls the p^{th} order average, as p increases the more unforgiving the OSPA becomes to outliers [27, 28, 32].

Oversimplifying somewhat, the OSPA searches for permutation of subsets of X and Y which are a minimal distance apart, penalizing any large differences in cardinality between X and Y. If either X or Y is empty then, from the OSPA's definition, the sets are said to be infinitely far apart. If both X and Y are empty, they are equal and the distance between them is zero.

The OSPA corrects the previous metrics' failings and is mathematically consistent, providing a global measure of the filter's accuracy. These metrics are not necessarily easy to interpret without the aid of additional performance measures, such as a comparison of the estimated RFS's cardinality to the ground truth.

5. Results

The UK-PHD was tested on both simulated data and measured data. The measured data was captured in February 2016 at the Eagle Canyon Driving Range. The simulated data attempts to replicate this same Eagle Canyon set up.

The three tees, or launch locations, used at the driving range were modelled as a constant Birth PHD added into the predicted PHD at every time step. The filter parameters were identical for both the simulated and measured tests, except for the measurement noise and detection probabilities. It was assumed the measured data is noisier with higher probabilities of missed detections.

On average the filter performed well, this section will just provide a small representative sample of the different types measured and simulated scenarios that were tested.

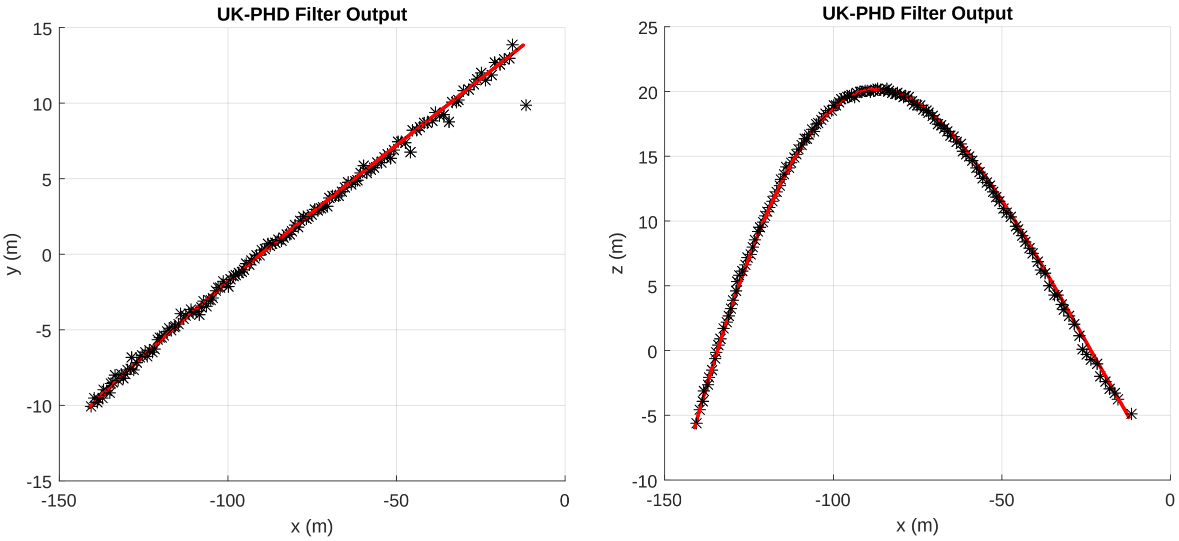
5.1 Simulated Data

The simulated data was tested with significantly less noise and a higher probability of detection than the measured data. The simulation of the golf ball trajectories and noise is immensely realistic; however it cannot account for external activity such as bird flight, movement on the range, etc.

There were several test cases most involving multiple targets in heavy clutter. Due to several, frustrating issues with time representation of the simulated measurements and ground truth, the OSPA will only be used as a performance measure on actual measured data.

5.1.1 Test Case 6

Test Case 6 is a near ideal scenario, a single target in a lower clutter environment in full sensor view. Figure 5.1 compares the estimated state's position with the ground truth, demonstrating that the UK-PHD is accurate under these conditions. Figure 5.2 plots the cardinality of the



(a) UK-PHD's estimated x-y trajectories.

(b) UK-PHD's estimated x-z trajectories.

Figure 5.1: The estimated UK-PHD trajectory compared to the ground truth for Test Case 6. The black stars represent the UK-PHD output, the red line the ground truth trajectory.

estimate RFS, the expected number of targets, over time. The instabilities in target number are caused by missed detections when using a higher probability of detection.

5.1.2 Test Case 9

Test case 9 is a near ideal multi-target scenario in low clutter, three targets are launched simultaneously from distinct locations.

Figure 5.3 compares the estimated states' position with the ground truth trajectories. As can be seen in Figure 5.3a, the estimated green trajectory is missing a significant portion. The target may have moved out of the sensors' view, resulting in missed detection the results of which are compounded by the IU. Its components are not necessary missing from the UK-PHD mixture, but they are not extracted as their weights are below the required threshold.

The expected number of targets, Figure 5.4, is accurate, but still subject to the same instabilities of Test Case 6. Sadly, due to time limitations, the cardinality is only plotted representation of the real time evolution of the system. The target's did indeed die at the times implied by Figure 5.4.

5.2 Measured Data with Performance Metrics

This section uses actual measured data from the demonstrator system used at Eagle Canyon in February 2016. Over 50 pin shots were recorded using all of the three launch locations, these are single shots where a player aims at specific flag, or pin, downrange.

The fitted trajectories from the demonstrator system, which are known to be accurate, will be used as the ground truth. The measurement noise has been increased significantly and the detection probability reduced, in attempt to make the filter robust.

Unfortunately due to time constraints, the estimated cardinality is the only measure of the real time evolution of the system. Only the OSPA will be used a multi-target miss distance, however the Hausdorf and OMAT metrics are provided in the implementation.

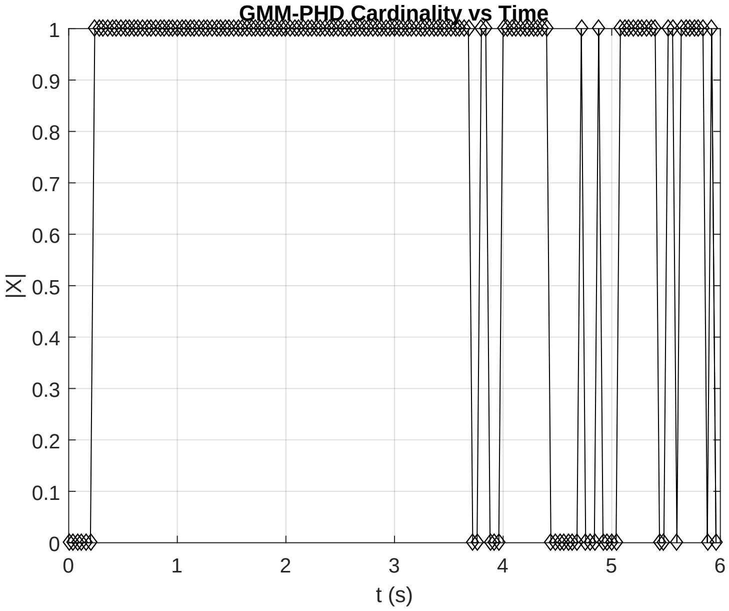
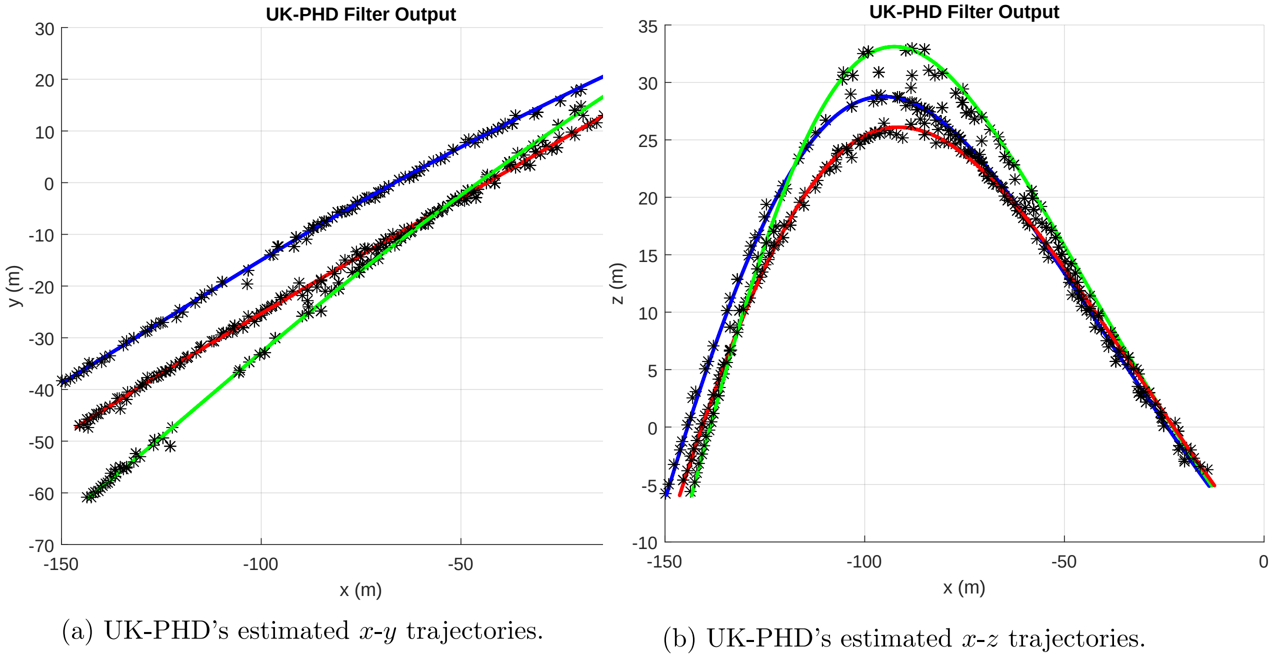


Figure 5.2: The cardinality of Test Case 6's measured RFS. The instabilities in the expected number of targets are a result of missed detections.



(a) UK-PHD's estimated x-y trajectories.

(b) UK-PHD's estimated x-z trajectories.

Figure 5.3: The estimated UK-PHD trajectory compared to the ground truth. The UK-PHD misses a significant portion of the green trajectory, this could be due to the target moving out of the sensors' view.

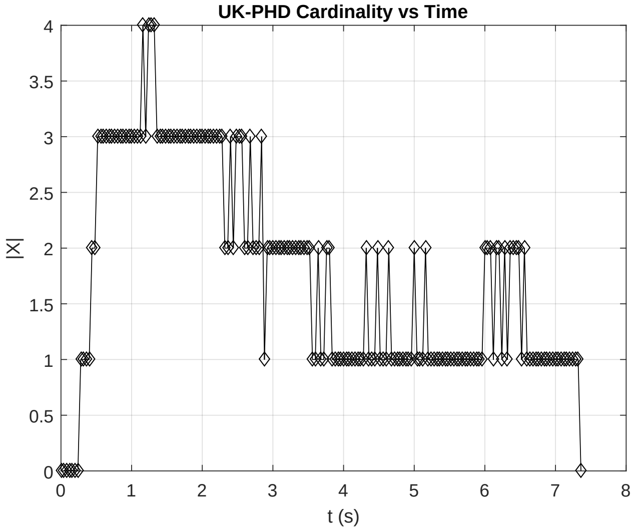


Figure 5.4: The cardinality of Test Case 9's measured RFS. The expected number of targets is still subject to the instabilities caused by missed detections while having a high detection probability.

5.2.1 Pin Shot 1

Pin shot 1 is an ideal scenario, there is only a single target in range and immensely low clutter. Figure 5.5 compares the UK-PHD estimate states' position to the fitted trajectory. The initial portion of all target's trajectories were too low to be seen by the Eagle Canyon detectors, the missing interval can be clearly seen in Figure 5.5b.

The OSPA and cardinality shown in Figure 5.6 are nearly perfect. The expected number of targets, , Figure 5.6a, is stable due to low clutter and a lower, more representative, detection probability.

5.2.2 Pin Shot 2

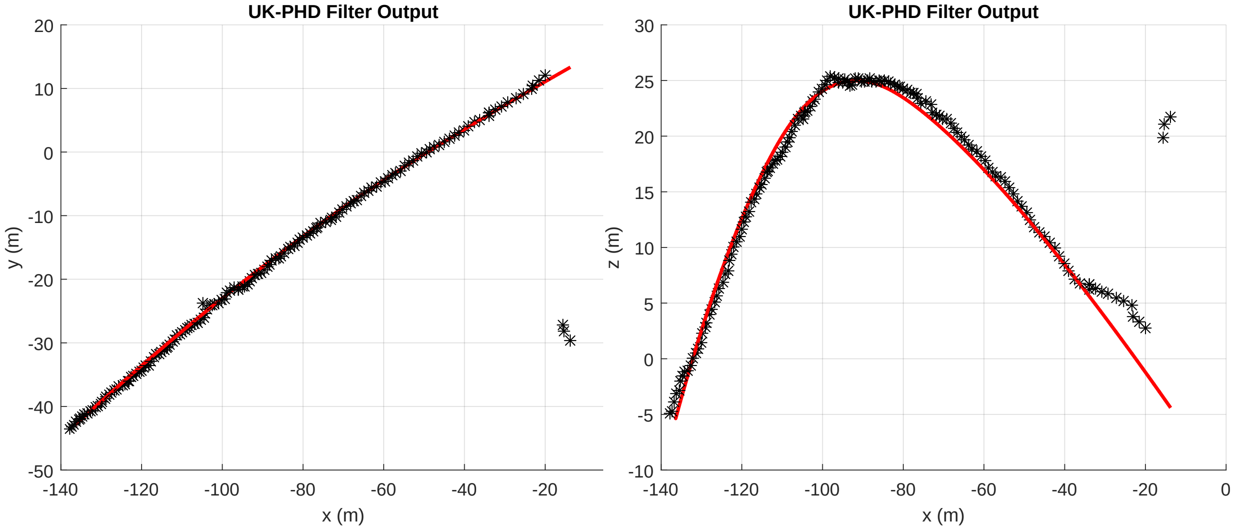
Pin Shot 2 is an absolute worst case scenario. Figure 5.7 shows that there is only a single true target, but two other golf balls are being hit from the side at the beginning of the measurements. These superfluous trajectories are still in the legal direction and therefore not removed. There is also an immense amount of clutter, although its origin is unknown. Figure 5.7b shows that the UK-PHD eventually does eventually track the target accurately, after missing a significant portion of the true trajectory due to initial tracking the false targets.

The estimated cardinality of Figure 5.8a is still fairly accurate, as it initially expects 3 targets on range. Figure 5.8b is high as the filter misses a significant portion of the actual trajectory while tracking the clutter.

5.2.3 Pin Shot 26

Pin Shot 26 is a single target in heavy clutter, chosen as it best represents the average pin shot. Figure 5.9 shows that the UK-PHD tracks the target accurately, without picking up to much false targets.

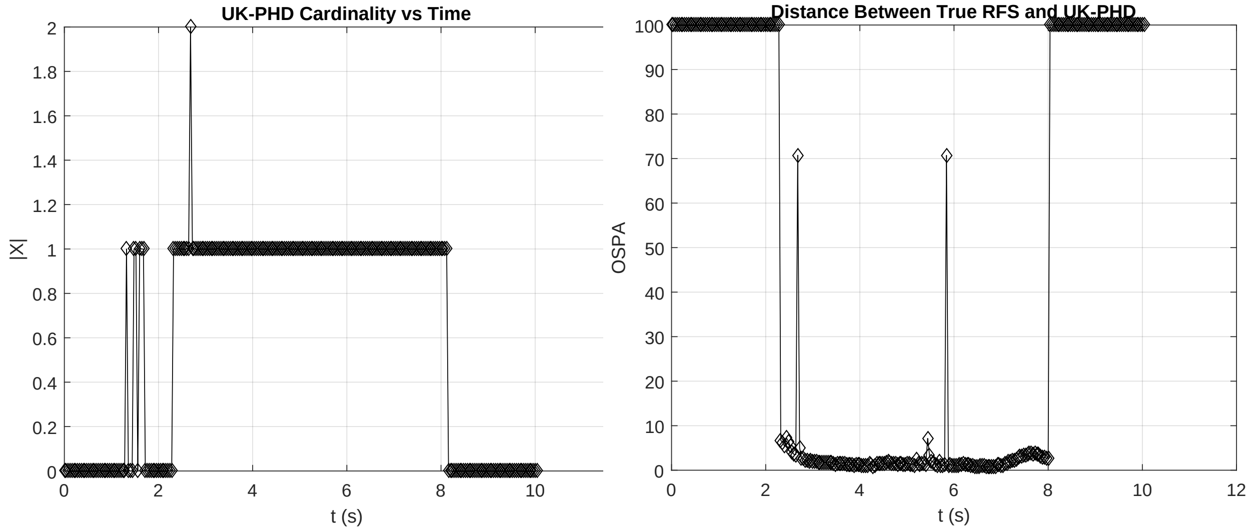
The estimated cardinality of Figure 5.8a is accurate, with minor instabilities caused by the clutter. Figure 5.8b also demonstrates the filter's accuracy in heavy clutter.



(a) UK-PHD's estimated x-y trajectories.

(b) UK-PHD's estimated x-z trajectories.

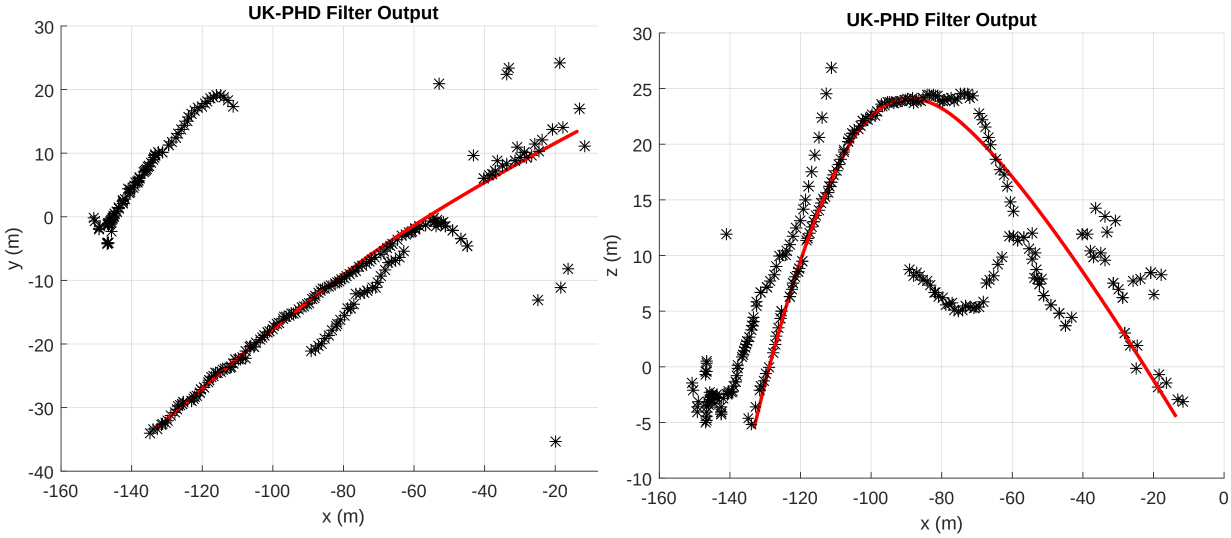
Figure 5.5: The UK-PHD's estimated target trajectories for Pin Shot 1. It can be seen clearly in Figure 5.9a that the initial portion of the estimated trajectory is missing.



(a) Estimated RFS cardinality, the expected number of targets.

(b) The OSPA for measured data, the distance between the ground truth and estimated RFSs.

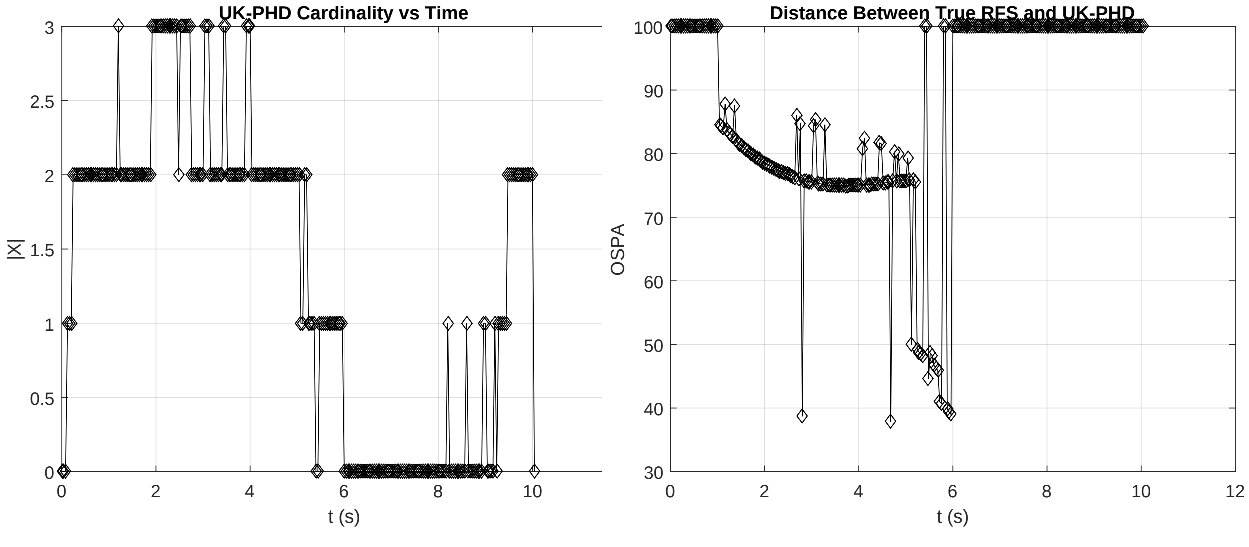
Figure 5.6: Performance metrics for Pin Shot 1. Both the OSPA and cardinality are nearly ideal. The stability of the expected target is due to a more representative detection probability.



(a) UK-PHD's estimated x-y trajectories.

(b) UK-PHD's estimated x-z trajectories.

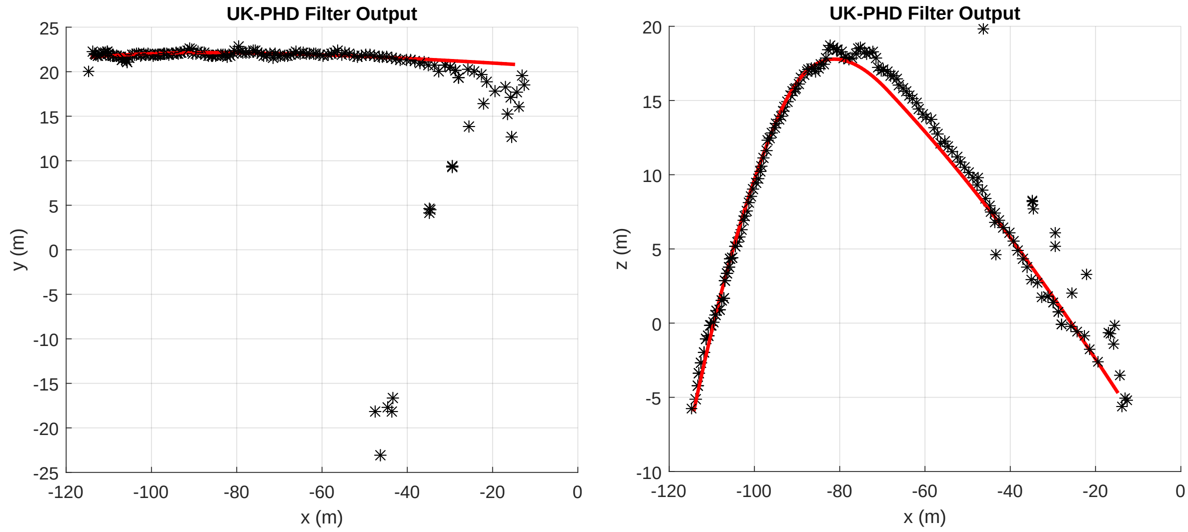
Figure 5.7: The UK-PHD's estimated target trajectories for Pin Shot 2. The UK-PHD filter struggles to track the true target when additional targets are present.



(a) Estimated RFS cardinality, the expected number of targets.

(b) The OSPA for measured data, the distance between the ground truth and estimated RFSs.

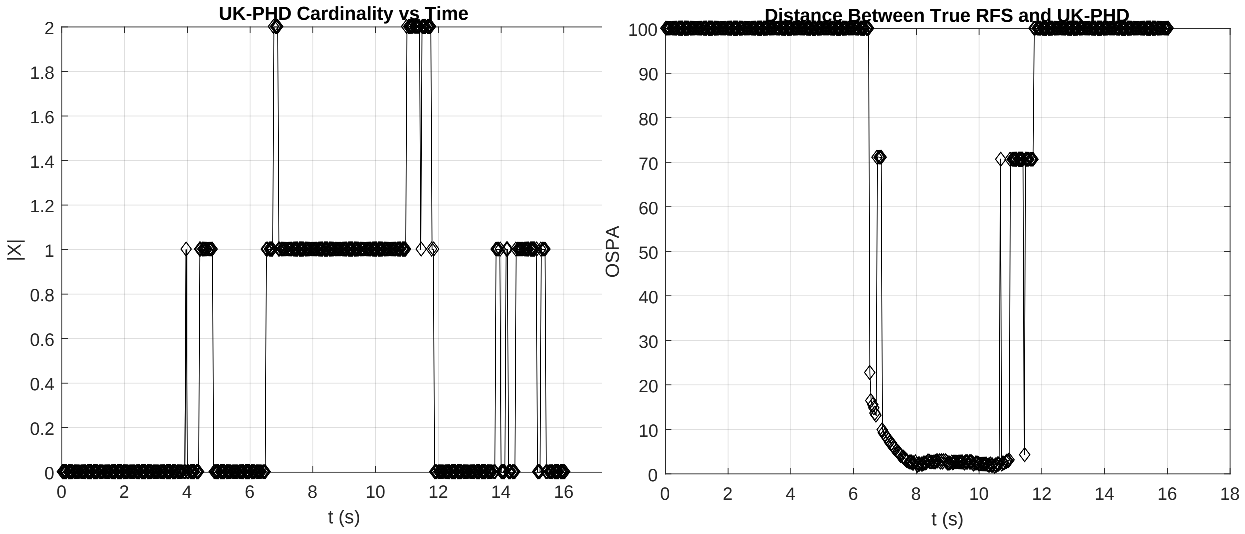
Figure 5.8: Performance metrics for Pin Shot 2. The expected number of targets is still accurate according to evolution of the system. The OSPA confirms that the filter is tracking false targets while inconsistently tracking the true target.



(a) UK-PHD's estimated x-y trajectories.

(b) UK-PHD's estimated x-z trajectories.

Figure 5.9: The UK-PHD's estimated target trajectories for Pin Shot 26. It can be seen clearly in Figure 5.9a that the initial portion of the estimated trajectory is missing.



(a) Estimated RFS cardinality, the expected number of targets.

(b) The OSPA for measured data, the distance between the ground truth and estimated RFSs.

Figure 5.10: Performance metrics for Pin Shot 26. The UK-PHD provides an accurate estimate of the true targets' states.

6. Conclusions and Future Work

It done work good.

Future work could include:

1. Finding a PGM representation of a RFS multi-target posterior. At a surface level, a model selection process could be presented as similar to a RFS. A discrete distribution describes the probability of a model, while families of continuous joint distributions describe the random vectors within.

2. Implementing an integrated method of labelling the tracks. [28] describes a maximum likelihood track association, while perhaps not accurate, will be sufficient. Other approaches could include labelling the mixture components [28], similar to labelled particles in the SMC-PHD.

3. Implementing local measures of accuracy, such as the Circular Position Error Probability (CPEP), which measure a single target's accuracy with its corresponding ground truth trajectory. The CPEP provides an accurate measure of track loss [13].

4. Implementing state dependent survival and detection probabilities. This was naively attempted, but it significantly complicated the expected target calculations. Done properly, it could allow targets to survive longer when moving in and out of the sensors' view.

References

[1] S. Robertson. ‘The Unscented Transform’. Internet: http://scjrobertson.github.io/documentation/ukf.pdf, 2016 [Jul. 14, 2016].

[2] J. Du Preez. ‘Inference: The Junction-Tree Algorithm’. Internet: http://research.ee.sun.ac.za/pgms/instructions/week5_junctiontree.pdf, 2016 [Apr. 26, 2016].

[3] B. Vo. ‘Codes’. Internet: http://ba-tuong.vo-au.com/codes.html, 2016 [Nov. 28, 2016].

[4] Zoeter et al. ‘Improved Unscented Kalman Smoothing for Stock Volatility Estimation’, presented at the 2004 IEEE Workshop on Machine Learning for Signal Processing, IEEE, 2004.

[5] S. Julier. “The Scaled Unscented Transformation”, in *Proc. IEEE Cat. No.CH37301*, 2002, pp. 4555 - 4559.

[6] S. Julier and J. Ulhmann. “A New Extension of the Kalman Filter to Nonlinear Systems”, in *Proc. SPIE 3068*, 1997.

[7] S. Julier and J. Ulhmann. “A General Method for Approximating Nonlinear Transformations of Probability Distributions”, Technical report, 1996.

[8] R. Mahler. “Multitarget Bayes filtering via first-order multitarget moments”. *IEEE Transactions on Aerospace and Electronic systems*, vol 39.4, pp. 1152-1178, 2003.

[9] J. Houssineau, E. Delande, D. Clark. “Notes of the summer school on finite set statistics.”. arXiv preprint arXiv:1308.2586 , 2013.

[10] R. Mahler. ““Statistics 101” for multisensor, multitarget data fusion”. *IEEE Aerospace and Electronic Systems Magazine*, vol 19.1, pp. 53-64, 2004.

[11] R. Mahler. ““Statistics 102” for multisource-multitarget detection and tracking”. *IEEE Journal of Selected Topics in Signal Processing*, vol 7.3, pp. 376-389, 2013.

[12] O. Erdinc, P Willet and Y. Bar-Shalom. “The Bin-Occupancy Filter and Its Connection to the PHD Filters.” *IEEE Transactions on Signal Processing*, vol. 57.11, pp. 4232-4246, 2009.

[13] B. N. Vo and W. K. Ma. “The cardinalized probability hypothesis density filter for linear Gaussian multi-target models.” *IEEE Transactions on Signal Processing*, vol. 54.11, pp. 4091-4104, 2006.

[14] B. N. Vo, B. T. Vo and A. Cantoni. “The Gaussian Mixture Probability Hypothesis Density Filter.” presented at 2006 40th Annual Conference on Information Sciences and Systems, IEEE, 2006.

[15] B. N. Vo, B. T. Vo and A. Cantoni. “The cardinality balanced multi-target multi-Bernoulli filter and its implementations.” *IEEE Transactions on Signal Processing*, vol. 57.2, pp. 409-423, 2009.

[16] B. N. Vo, B. T. Vo and D. Phung. "Labeled random finite sets and the Bayes multi-target tracking filter." *IEEE Transactions on Signal Processing*, vol. 62.24, pp. 6554-6567, 2014.

[17] Panta, Kusha, Ba-Ngu Vo, and Sumeetpal Singh. "Novel data association schemes for the probability hypothesis density filter." *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43.2, pp. 556-570, 2007.

[18] Li, Wenling, et al. "Gaussian mixture PHD filter for multi-sensor multi-target tracking with registration errors." Signal Processing, vol. 93.1, pp. 86-99, 2013.

[19] Liu, Long, Hongbing Ji, and Zhenhua Fan. "Improved Iterated-corrector PHD with Gaussian mixture implementation." *Signal Processing*, vol 115, pp. 82-100, 2015.

[20] Yazdian-Dehkordi, Mahdi, and Zohreh Azimifar. "Refined GM-PHD tracker for tracking targets in possible subsequent missed detections." *Signal Processing*, vol 116, pp. 112-126, 2015.

[21] Xu, Jian, Fang-ming Huang, and Zhi-liang Huang. "The multi-sensor PHD filter: Analytic implementation via Gaussian mixture and effective binary partition." in *Information Fusion (FUSION), 2013 16th International Conference on*, 2013, pp. 945-952.

[22] Nagappa, Sharad, and Daniel E. Clark. "On the ordering of the sensors in the iterated-corrector probability hypothesis density (PHD) filter." in *SPIE Defense, Security, and Sensing*, 2011, pp. 80500M–80500M.

[23] R. Mahler. "Approximate multisensor CPHD and PHD filters." in *Information Fusion (FUSION), 2010 13th Conference on*, 2010, pp. 1-8.

[24] R. Mahler. 'A brief survey of advances in random-set fusion', presented at International Conference on Control, Automation and Information Sciences (ICCAIS), IEEE, 2015.

[25] R. Mahler. 'Integral-transform derivations of exact closed-form multitarget trackers', presented at 19th International Conference on Information Fusion (FUSION), ISIF, 2016.

[26] J. Hoffman and R. Mahler. "Multitarget miss distance and its applications." Proceedings of the Fifth International Conference on Information Fusion, 2002, pp. 149-155.

[27] D. Schuhmacher, B. Vo and B. Vo. "A consistent metric for performance evaluation of multi-object filters." *IEEE Transactions on Signal Processing*, vol. 56.8, pp. 3447-3457, 2008.

[28] M. Fröhle. "Multi-target Tracking for UWB Channels Using PHD Filters." MSc thesis, Graz University of Technology, Austria, 2011.

[29] T. Minka. 'A Family of Algorithms for Approximate Bayesian Inference'. PhD thesis, M.I.T, U.S.A, 2001.

[30] M. Vihola. "Random sets for multitarget tracking and data fusion." Licentiate Thesis, Tampere University of Technology, Italy, 2004.

[31] K. Panta. "Multi-Target Tracking Using 1st Moment of Random Finite Sets." PhD thesis, University of Melbourne, Australia, 2007.

[32] B. Vo. "Random finite sets in multi-object filtering." PhD Thesis, University of Western Australia, Australia, 2008.

[33] T. Wood. "Random Finite Sets for Multitarget Tracking with Applications." PhD thesis, University of Oxford, U.K, 2011.

[34] M. Bocquel. "Random Finite Sets in Multi-target Tracking: Efficient Sequential MCMC implementation." PhD Thesis, University of Twente, Netherlands, 2013.

[35] A. Gilat and V. Subramaniam. 'Numerical Integration' in *Numerical Methods: An Introduction with Applications using MATLAB*. M. McDonald, 2$^{\text{nd}}$ ed., Columbus, Ohio: John Wiley & Sons, 2011, pp. 263-269.

[36] H. Nguyen. 'Finite Random Sets' in *An Introduction to Random Sets*, CRC press, 2006, pp. 35-70.

[37] R. Mahler. 'Random Set Theory for Target Tracking and Identification' in *Handbook of Multisensor Data Fusion*, CRC press, 2001, pp. 91-194.

[38] E. Biglieri, E. Grossi, and M. Lops. 'An Engineering Introduction' in *Random-Set Theory and Wireless Communications*, Now, 2012, pp. 323-341.

[39] I. Goodman et al. 'The Random Set Approach to Data Fusion' in *Mathematics of Data Fusion*, 1$^{\text{st}}$ ed., vol. 1. Springer Science & Business Media, 2013, pp. 91-194.

[40] I. Goodman et al. 'Fusion of Unambiguous Data' in *Mathematics of Data Fusion*, 1$^{\text{st}}$ ed., vol. 1. Springer Science & Business Media, 2013, pp. 219-261.

[41] D. Barber. 'Belief Networks' in *Bayesian Reasoning and Machine Learning*, 1$^{\text{st}}$ ed., vol. 1. Cambridge, U.K: Cambridge University Press, 2012, pp. 29-58.

[42] D. Barber. 'Graphical Models' in *Bayesian Reasoning and Machine Learning*, 1$^{\text{st}}$ ed., vol. 1. Cambridge, U.K: Cambridge University Press, 2012, pp. 58-77.

[43] D. Barber. 'The Junction Tree Algorithm' in *Bayesian Reasoning and Machine Learning*, 1$^{\text{st}}$ ed., vol. 1. Cambridge, U.K: Cambridge University Press, 2012, pp. 102-127.

[44] D. Koller and N. Friedman. 'The Bayesian Network Representation' in *Probabilistic Graphical Models: Principles and Techniques*, 1$^{\text{st}}$ ed., vol. 1. Cambridge, Massachusetts: The MIT Press, 2009, pp. 45-83.

[45] D. Koller and N. Friedman. 'Exact Inference' in *Probabilistic Graphical Models: Principles and Techniques*, 1$^{\text{st}}$ ed., vol. 1. Cambridge, Massachusetts: The MIT Press, 2009, pp. 345-376.

[46] D. Koller and N. Friedman. 'Inference in Hybrid Networks' in *Probabilistic Graphical Models: Principles and Techniques*, 1$^{\text{st}}$ ed., vol. 1. Cambridge, Massachusetts: The MIT Press, 2009, pp. 609-611.

[47] S. Thrun et al. 'Recursive State Estimation' in *Probabilistic Robotics*, 1$^{\text{st}}$ ed., vol. 1. Cambridge, Massachusetts: The MIT Press, 2009, pp. 14-35.

[48] S. Thrun et al. 'Gaussian Filters' in *Probabilistic Robotics*, 1$^{\text{st}}$ ed., vol. 1. Cambridge, Massachusetts: The MIT Press, 2009, pp. 39-53.