The Cardinality Balanced Multi-Target Multi-Bernoulli Filter and its Implementations

Ba-Tuong Vo, Ba-Ngu Vo, and Antonio Cantoni

Abstract—It is shown analytically that the Multi-Target Multi-Bernoulli (MeMBer) recursion, proposed by Mahler, has a significant bias in the number of targets. To reduce the cardinality bias, a novel multi-Bernoulli approximation to the multi-target Bayes recursion is derived. Under the same assumptions as the MeMBer recursion, the proposed recursion is unbiased. In addition, a Sequential Monte Carlo (SMC) implementation (for generic models) and a Gaussian mixture (GM) implementation (for linear Gaussian models) are proposed. The latter is also extended to accommodate mildly non-linear models by linearization and the unscented transform.

Index Terms—Tracking, estimation, random sets, point processes, finite set statistics, multi-Bernoulli.

I. INTRODUCTION

Multi-target tracking, involves the joint estimation of the number of targets and their individual states from a sequence of observations in the presence of detection uncertainty, association uncertainty and clutter [1]–[3]. Mahler’s finite set statistics (FISST) is an elegant Bayesian formulation of multi-target filtering based on random finite set (RFS) theory, which has generated substantial interest in recent years due to the development of the Probability Hypothesis Density (PHD) filter [4] and the Cardinalized PHD (CPHD) filter [5]. The PHD and CPHD filters are moment approximations of the Bayes multi-target filter, which operate on the single-target state space and avoid the combinatorial problem that arises from data association. Sequential Monte Carlo (SMC) implementations [6]–[8] with provable convergence [6], [9], [10] and closed form solutions [11], [12] have opened the door to numerous novel extensions and applications which can be found in the surveys [3], [13].

In addition to the PHD and CPHD filters, Mahler recently proposed the Multi-Target Multi-Bernoulli (MeMBer) recursion as a tractable approximation to the Bayes multi-target recursion under low clutter density scenarios [3]. Unlike the PHD and CPHD recursions, which propagate moments and cardinality distributions, the MeMBer recursion propagates (approximately) the multi-target posterior density. Specifically, the parameters of a multi-Bernoulli RFS that approximates the posterior multi-target RFS is derived. Under the same assumptions as the MeMBer recursion, the proposed recursion is unbiased. In addition, a Sequential Monte Carlo (SMC) implementation (for generic models) and a Gaussian mixture (GM) implementation (for linear Gaussian models) are proposed. The latter is also extended to accommodate mildly non-linear models by linearization and the unscented transform.

In this paper, we show analytically that Mahler’s MeMBer filter over-estimates the cardinality (number of targets) and propose a novel filter that is unbiased in cardinality under the same signal setting. Specifically, we derive the cardinality bias that occurs in the MeMBer data update step, and use this to develop an unbiased update. The proposed filter, called the cardinality-balanced MeMBer (CBMeMBer) filter, propagates a set of multi-Bernoulli parameters characterizing the posterior multi-target RFS. Sequential Monte Carlo techniques for the standard Bayes [14] and PHD/CPHD [6], [15] recursions cannot be utilized directly for the CBMeMBer (nor MeMBer) recursion, and at present, no general implementation exists.

We propose a generic Sequential Monte Carlo (SMC) implementation of the CBMeMBer recursion (also applicable to the MeMBer recursion) that accommodates nonlinear dynamic and measurement models with state-dependent sensor field of view. The key advantage of this approach over the SMC-PHD filters is that the multi-Bernoulli representation allows reliable and inexpensive extraction of state estimates. In contrast, the SMC-PHD approach requires clustering to extract state estimates from the particle population, which is expensive and unreliable [11]. We also propose a Gaussian mixture (GM) implementation for linear Gaussian multi-target models and extend this technique to mildly nonlinear multi-target models (with uniform sensor field of view) via linearization and the unscented transform. Simulations demonstrate significant reduction of false tracks, under harsher signal settings than initially assumed. Our numerical studies also show that under certain range of signal settings, the SMC-CBMeMBer filter outperforms the SMC-CPHD (and hence SMC-PHD) filter, despite having smaller complexity. The GM-CBMeMBer filter though, is only comparable to the GM-PHD filter.

Preliminary results have been announced in the conference paper [16]. This paper presents a more complete analytical and numerical study. In summary, our contributions are:

- an analytic expression for the cardinality bias in the MeMBer filter;
- the CBMeMBer filter;
- a generic SMC implementation for general nonlinear multi-target models;
- analytic implementations for linear and mildly nonlinear multi-target models;
- performance evaluation against the PHD/CPHD filters.
The paper is organised as follows. The necessary background on RFSs and multi-target filtering is given in Section II. Section III presents a review of Mahler’s MeMBer recursion, followed by a derivation of the cardinality bias, and the CBMeMBer update. A generic Sequential Monte Carlo implementation for non-linear multi-target models is described in Section IV-A and analytic Gaussian mixture implementations for linear and mildly non-linear multi-target models are given in Section IV-B. Numerical studies are shown in Section V. Closing remarks are given in Section VI.

II. BACKGROUND

This section introduces RFSs, multi-target system models and the multi-target Bayes filter.

A. Random Finite Sets

Intuitively, a random finite set (RFS), is a random (spatial) point pattern, e.g. measurements on a radar screen. What distinguishes an RFS from a random vector is that: the number of points is random; the points themselves are random and unordered. In essence, an RFS is simply a finite-set-valued random variable. At the fundamental level, like any other random variable, the randomness of an RFS is captured by its probability distribution or probability density.

Another fundamental descriptor of an RFS, which has direct relevance to this paper, is the probability generating functional (PGFI). Let $\mathcal{F}(X)$ denote the space of finite subsets of $X \subseteq \mathbb{R}^n$. Suppose $X$ is an RFS on $X$, i.e. $X$ is a random variable taking values in $\mathcal{F}(X)$. Following [17], [18], the probability generating functional (PGFI) $G[\cdot]$ of $X$ is defined by

$$G[h] \equiv \mathbb{E}[h^X],$$

where $\mathbb{E}$ denotes the expectation operator, $h$ is any real-valued function on $X$ such that $0 \leq h(x) \leq 1$, and $h^X \equiv \prod_{x \in X} h(x)$ with $h^\emptyset = 1$ by convention. The cardinality (number of elements) of $X$, denoted as $|X|$, is a discrete random variable whose probability generating function $G(\cdot)$ can be obtained by substituting the constant function $h(x) = y$ into the PGFI. Some examples of RFS pertinent to development of our key results are given next.

1) Poisson RFS: An RFS $X$ on $X$ is said to be Poisson with a given intensity function $\nu$ (defined on $X$) if its cardinality is Poisson distributed, with mean $\bar{N} = \int \nu(x)dx$, and for any finite cardinality, the elements $x$ of $X$ are independently and identically distributed (i.i.d.) according to the probability density $\nu(\cdot)/\bar{N}$ [17], [18]. A Poisson RFS is completely characterized by its intensity function, also known in the tracking literature as the Probability Hypothesis Density (PHD). A Poisson RFS with intensity function $\nu$ has PGFI $G[h] = e^{\nu(h-1)}$, where $\langle \nu, h \rangle = \int \nu(x)h(x)dx$ (see [3] pp. 374). The probability density $1$ of a Poisson RFS can also be explicitly expressed in terms of $\nu$ as $\pi(X) = e^{-\bar{N}}\bar{N}^X$ (see [3] pp. 366).

$^1$For simplicity, in this paper, we shall not distinguish a FISST set derivative of a belief functional and a probability density. While the former is not a probability density [3], it is, equivalent to a probability density relative to the distribution of a Poisson RFS with unit intensity (see [6]).

2) Bernoulli RFS: A Bernoulli RFS on $X$ has probability $1-r$ of being empty, and probability $r$ of being a singleton whose (only) element is distributed according to a probability density $p$ (defined on $X$). The cardinality distribution of a Bernoulli RFS is a Bernoulli distribution with parameter $r$. The PGFI of a Bernoulli RFS is (see [3] pp. 375)

$$G[h] = 1 - r + r(p, h),$$

while its probability density is (see [3] pp. 368)

$$\pi(X) = \begin{cases} 1-r & X = \emptyset, \\ r \cdot p(x) & X = \{x\}. \end{cases}$$

3) Multi-Bernoulli RFS: A multi-Bernoulli RFS $X$ on $X$ is a union of a fixed number of independent Bernoulli RFSs $X^{(i)}$ with existence probability $\nu^{(i)} \in (0, 1)$ and probability density $p^{(i)}$ (defined on $X$), $i = 1, \ldots, M$, i.e. $X = \bigcup_{i=1}^M X^{(i)}$. The PGFI of a multi-Bernoulli RFS is (see [3] pp. 375)

$$G[h] = \prod_{i=1}^M \left( 1 - r^{(i)} + r^{(i)}(p^{(i)}, h) \right).$$

A multi-Bernoulli RFS is thus completely described by the multi-Bernoulli parameter set $\{\nu^{(i)}, p^{(i)}\}_{i=1}^M$. The mean cardinality of a multi-Bernoulli RFS is $\sum_{i=1}^M r^{(i)}$. Moreover, the probability density $\pi$ is (see [3] pp. 368) $\pi(\cdot) = \prod_{i=1}^M (1 - r^{(i)})$ and

$$\pi(\{x_1, \ldots, x_n\}) = \pi(\emptyset) \sum_{1 \leq i_1 < \ldots < i_n \leq M} \frac{r^{(i_1)} \cdots r^{(i_n)} p^{(i_1)}(x_{i_1}) \cdots p^{(i_n)}(x_{i_n})}{1 - r^{(i_1)} \cdots r^{(i_n)}}.$$ 

Throughout this paper, we abbreviate a probability density of the form (5) by $\pi = \{r^{(i)}, p^{(i)}\}_{i=1}^M$. We also refer to a PGFI of the form (4), or a multi-target density of the form (5), as multi-Bernoulli.

B. Multi-Target System Model

Suppose that at time $k$, there are $N(k)$ target states $x_{k,1}, \ldots, x_{k,N(k)}$, each taking values in a state space $X \subseteq \mathbb{R}^n$, and $M(k)$ measurements $z_{k,1}, \ldots, z_{k,M(k)}$ each taking values in an observation space $Z \subseteq \mathbb{R}^n$. In the random finite set approach, the finite sets of targets and observations, at time $k$, [3], [4], [19] are treated as the multi-target state and multi-target observation, respectively

$$X_k = \{x_{k,1}, \ldots, x_{k,N(k)}\} \in \mathcal{F}(X),$$

$$Z_k = \{z_{k,1}, \ldots, z_{k,M(k)}\} \in \mathcal{F}(Z).$$

Using RFS theory, we can construct stochastic models for the time evolution of the multi-target state and the multi-target observations as follows.

Given a multi-target state $X_{k-1}$ at time $k-1$, each $x_{k-1} \in X_{k-1}$ either continues to exist at time $k$ with probability $p_{S,k}(x_{k-1})$ and move to a new state $x_k$ with probability density $f_{k|k-1}(x_k|x_{k-1})$, or dies with probability $1 - p_{S,k}(x_{k-1})$. Thus, a given target with state $x_{k-1} \in X_{k-1}$ at time $k-1$, its behaviour time $k$ is modeled by the Bernoulli RFS $S_{k|k-1}(x_{k-1})$ with $r = p_{S,k}(x_{k-1})$ and $p(\cdot) = f_{k|k-1}(\cdot|x_{k-1})$. This transition is commonly known in
Thus, at time $k$, modeled as a Poisson RFS $r$ with using multi-Bernoulli RFSs. Intuitively, the MeMBer recursion approximates to the full multi-target Bayes recursion (8-9) integrals (see [3], [4], [19]),

$$Z_k = \bigcup_{x \in X_k} \Theta_k(x) \cup K_k.$$  \hfill (7)

The RFS multi-target measurement equation (7) accounts for detection uncertainty and clutter. It is assumed that the RFSs constituting the union in (7) are independent of one another. Note that conditional on $X_k$, the target generated measurements in (7) form a multi-Bernoulli RFS.

### C. Multi-Target Bayes Recursion

The multi-target filtering problem can be posed as a Bayes filter with state space $\mathcal{F}(X)$ and observation space $\mathcal{F}(Z)$. Let $\pi_k(X|Z_{1:k})$ denote the multi-target posterior density at time $k$. Then, the multi-target Bayes recursion propagates $\pi_k(X|Z_{1:k})$ in time [4], [6], [19] according to

$$\pi_{k|k-1}(X_k|Z_{1:k-1}) = \int f_{k|k-1}(X_k|X_k) \pi_{k-1}(X|Z_{1:k-1}) dX,$$

$$\pi_k(X|Z_{1:k}) = \int g_k(Z_k|X_k) \pi_{k|k-1}(X_k|Z_{1:k-1}) dX,$$  \hfill (8)

where the integrals in the above recursion are FISST set integrals (see [3], [4], [19]), $f_{k|k-1}(|\cdot|)$ is the multi-target transition density and $g_k(\cdot|\cdot)$ is the multi-target likelihood. The multi-target transition encapsulates the underlying models of target motions, births and deaths described in (6), while the multi-target likelihood encapsulates the underlying models of detections and false alarms described in (7). Explicit expressions for $f_{k|k-1}(X_k|X_{k-1})$ and $g_k(Z_k|X_k)$ can be derived from (6) and (7) using FISST, see for example [3], [4], [19].

### III. MeMBer Approximations

The MeMBer recursion, proposed by Mahler in [3], is an approximation to the full multi-target Bayes recursion (8-9) using multi-Bernoulli RFSs. Intuitively, the MeMBer recursion propagates the multi-target posterior probability density in time by propagating a finite but time-varying number of hypothesized tracks, each characterized by the probability of existence and the probability density of the current hypothesized state.

In this section, we briefly summarize Mahler’s MeMBer recursion and derive the cardinality bias that occurs in the update step. Moreover, we propose a MeMBer update step that is unbiased in cardinality, herein referred to as the cardinality-balanced MeMBer (CBMeMBer) update.

#### A. The Original MeMBer Recursion

The premise of the MeMBer recursion is that the multi-target RFS at each time step is approximated by a multi-Bernoulli RFS, based on the following modelling assumptions:

- Each target evolves and generates measurements independently.
- Target births follow a multi-Bernoulli RFS independent of target survivals.
- Clutter follows a Poisson RFS, not too dense, and is independent of target-generated measurements.

The MeMBer recursion is summarized in Propositions 1 and 2 as follows. The original equations are found in [3], for the prediction on pp. 661 eqs. (17.36)-(17.43), and for the update on pp. 662 eqs. (17.45)-(17.55).

**Proposition 1 (MeMBer Prediction).** If at time $k - 1$, the posterior multi-target density is a multi-Bernoulli of the form

$$\pi_{k-1} = \{ (r_{k-1}^{(i)}, P_{k-1}^{(i)}) \}_{i=1}^{M_{k-1}},$$

then the predicted multi-target density is also a multi-Bernoulli and is given by

$$\pi_{k|k-1} = \{ (r_{k|k-1}^{(i)}, P_{k|k-1}^{(i)}) \}_{i=1}^{M_k},$$

where

$$r_{k|k-1}^{(i)} = \frac{r_{k-1}^{(i)} P_{k-1}^{(i)} \pi_{k-1}^{(i|\cdot)}}{(p_{k-1}^{(i)} P_{k-1}^{(i)} \pi_{k-1}^{(\cdot|\cdot)})},$$

$$P_{k|k-1}^{(i)}(x) = \frac{\{ f_{k|k-1}^{(i)}(x|x) P_{k-1}^{(i)} \pi_{k-1}^{(\cdot|\cdot)} \}}{(p_{k-1}^{(i)} P_{k-1}^{(i)} \pi_{k-1}^{(\cdot|\cdot)})},$$

The statement of the MeMBer update in Proposition 2 is summarized as an approximate update in the interest of readability. The formal statement would otherwise require lengthy descriptions of its premises and consequently is not presented here. The complete statement is given in the original equations in [3].
The total number of predicted hypothesized tracks is $M_{k|k-1} = M_{k-1} + M_{T,k}$.

**Proposition 2 (MeMBer Update).** If at time $k$, the predicted multi-target density is a multi-Bernoulli of the form
\[
\pi_{k|k-1} = \left\{ \{r_{k|k-1}^{(i)}, p_{k|k-1}^{(i)}\} \right\}_{i=1}^{M_{k|k-1}},
\]
then the posterior multi-target density can be approximated by a multi-Bernoulli as follows
\[
\pi_k \approx \{r_{L,k}^{(i)}, p_{L,k}^{(i)}\} \cup \{(r_{U,k}(z), p_{U,k}(z); z)\} \in Z_k,
\] (13)
where
\[
\begin{align*}
r_{L,k}^{(i)} &= r_{k|k-1}^{(i)} \frac{1 - p_{k|k-1}^{(i)} h q_{D,k}}{1 - r_{k|k-1}^{(i)} h q_{D,k}}, \\
p_{L,k}^{(i)}(x) &= \frac{p_{k|k-1}^{(i)}(x)}{1 - p_{D,k}(x)}, \\
r_{U,k}(z) &= \sum_{i=1}^{M_{k|k-1}} r_{k|k-1}^{(i)} (p_{k|k-1}^{(i)}(z) \psi_{k,z}), \\
p_{U,k}(z; x) &= \sum_{i=1}^{M_{k|k-1}} r_{k|k-1}^{(i)} (p_{k|k-1}^{(i)}(z) \psi_{k,z}), \\
\psi_{k,z} &= g_k(z|x) p_{D,k}(x), \\
Z_k &= \text{measurement set at time } k, \\
g_k(\cdot|x) &= \text{single target measurement likelihood at time } k, \\
p_{D,k}(x) &= \text{probability of target detection at time } k \\
\kappa_k(\cdot) &= \text{intensity of Poisson clutter at time } k.
\end{align*}
\]

It is implicitly assumed that $p_{D,k}$ and $r_{k|k-1}^{(i)}$, $i = 1, ..., M_{k|k-1}$, cannot all be equal to 1. In essence, the multi-Bernoulli parameter set for the updated multi-target density $\pi_k$ is formed by the union of the multi-Bernoulli parameter sets for the legacy tracks (the first term in (13)) and measurement-corrected tracks (the second term in (13)). The total number of posterior hypothesized tracks is $M_k = M_{k|k-1} + |Z_k|$.

While the data prediction step of the MeMBer recursion is exact, the data update step is based on the following approximation to the PGFl of the posterior multi-target state at time $k$ (see [3] pp. 680 eqs. (17.173)-(17.174))
\[
G_{k|k}[h|z] = \prod_{i=1}^{M_{k|k-1}} G_{L,k}^{(i)}[h] \prod_{z \in Z_k} G_{U,k}(z; h),
\] (18)
where
\[
\begin{align*}
G_{L,k}^{(i)}[h] &= \frac{1 - r_{k|k-1}^{(i)} (p_{k|k-1}^{(i)} h q_{D,k})}{1 - r_{k|k-1}^{(i)} (p_{k|k-1}^{(i)} h q_{D,k})}, \\
G_{U,k}[h|z] &= \frac{\kappa_k(z) + \sum_{i=1}^{M_{k|k-1}} G_{U,k}^{(i)}[h|z]}{\kappa_k(z) + \sum_{i=1}^{M_{k|k-1}} G_{U,k}^{(i)}[1; z]},
\end{align*}
\] (19, 20)

This is a reasonable approximation when clutter is not too dense as assumed at the onset of this section. The reader is referred to Mahler’s original derivation [3] (see pp. 678-680) for the rationale behind this approximation.

Observe however that the first product in (18) is a multi-Bernoulli but the second product is not. In fact, each factor $G_{U,k}[h|z]$ of the second product might not even be a PGFl of an RFS. Nonetheless, finding a Bernoulli approximation to $G_{U,k}[h|z]$ allows the second product in (18) (and hence $G_k[\cdot]$) to be approximated by a multi-Bernoulli. In the original MeMBer update approximation, Mahler simply sets $h = 1$ in the denominator of (21), i.e.
\[
G_{U,k}^{(i)}[h|z] \approx \frac{r_{k|k-1}^{(i)} (p_{k|k-1}^{(i)} h q_{D,k})}{1 - r_{k|k-1}^{(i)} (p_{k|k-1}^{(i)} h q_{D,k})},
\] (22)

(see [3] pp. 681 eq. (17.176)) which, after substituting into (20), yields the following Bernoulli approximation
\[
G_{U,k}[h|z] \approx 1 - r_{U,k}(z) + r_{U,k}(z) (p_{U,k}(z; z), h),
\] (23)

with $r_{U,k}(z)$ and $p_{U,k}(z; z)$ given by (16) and (17) (see [3] pp. 681 eq. (17.184)). It is unclear why (23) is a good approximation. Moreover, it will be shown this particular approximation leads to a bias in the cardinality of the measurement-updated tracks and consequently a bias in the posterior cardinality.

**B. Cardinality Bias**

It follows from Proposition 1 that the mean cardinality of the predicted multi-target state is
\[
\bar{N}_{k|k-1} = \sum_{i=1}^{M_{k|k-1}} r_{k|k-1}^{(i)} + \sum_{i=1}^{M_{T,k}} r_{T,k}^{(i)}.
\]

According to Proposition 2, the posterior multi-target density is approximated by a multi-Bernoulli with mean cardinality
\[
\bar{N}_k = \sum_{i=1}^{M_{k|k-1}} r_{L,k}^{(i)} + \sum_{z \in Z_k} r_{U,k}(z).
\] (24)

This mean is not necessarily the mean cardinality of the posterior multi-target state (or simply the mean posterior cardinality) even if equality holds in approximation (18). The following result gives the mean posterior cardinality under the assumption that equality holds in (18).

**Proposition 3.** If the PGFl of the posterior multi-target density is
\[
G_k[h] = \prod_{i=1}^{M_{k|k-1}} G_{L,k}^{(i)}[h] \prod_{z \in Z_k} G_{U,k}(z; h),
\] (25)

then the mean cardinality of the posterior multi-target state at time $k$ is
\[
\bar{N}_k = \sum_{i=1}^{M_{k|k-1}} r_{L,k}^{(i)} + \sum_{z \in Z_k} r_{U,k}(z),
\] (26)
where \( r^{(i)}_{L,k} \) is given by (14), and

\[
\begin{align*}
\tau^*_U,k(z) &= \sum_{i=1}^{M_{L,k-1}} \frac{r^{(i)}_{L,k-1}(1-r^{(i)}_{L,k-1})}{1-r^{(i)}_{L,k-1}} \left( \frac{p^{(i)}_{L,k-1}(x|\theta_{k,z})}{r^{(i)}_{L,k-1} + p^{(i)}_{L,k-1}(x|\theta_{k,z})} \right)^2, \\
r_k(z) &= \sum_{i=1}^{M_{L,k-1}} \frac{r^{(i)}_{L,k-1}(1-r^{(i)}_{L,k-1})}{1-r^{(i)}_{L,k-1}} \left( \frac{p^{(i)}_{L,k-1}(x|\theta_{k,z})}{r^{(i)}_{L,k-1} + p^{(i)}_{L,k-1}(x|\theta_{k,z})} \right)^2.
\end{align*}
\]  
(27)

**Proof:** The product, \( \prod_{i=1}^{M_{L,k-1}} G_{L,k}^{(i)}[h] \), in (25), corresponds to the set of legacy tracks and is a multi-Bernoulli, since (19) can be rewritten in Bernoulli form

\[
G_{L,k}[h] = 1 - r^{(i)}_{L,k} + r^{(i)}_{L,k} p^{(i)}_{L,k}(x|\theta_{k,z})
\]

with \( r^{(i)}_{L,k} \) and \( p^{(i)}_{L,k} \) given as in Proposition 2. Hence the mean cardinality of the legacy tracks is

\[
\sum_{i=1}^{M_{L,k-1}} r^{(i)}_{L,k}.
\]  
(28)

The product, \( \prod_{z \in Z_k} G_{U,k}[h; z] \) in (25), corresponds to the set of measurement-updated tracks, which is not a multi-Bernoulli RFS. Nonetheless, the corresponding mean cardinality can be computed exactly. Indeed, the mean cardinality of the measurement-updated tracks is

\[
\sum_{z \in Z_k} G_{U,k}(1; z).
\]  
(29)

Substituting \( h(x) = y \) into the PGFI (20-21); then differentiating at \( y = 1 \) yields \( G_{U,k}(1; z) = \tau^*_U,k(z) \) as given in (27). It follows from (18) that the mean posterior cardinality is the sum of the legacy track cardinality (28) and measurement-updated track cardinality (29)

**Corollary to Proposition 3.** Under the premises of Proposition 3, the posterior cardinality bias at time \( k \) is

\[
\tilde{N}_k - \bar{N}_k = \sum_{z \in Z_k} \left( \frac{M_{L,k-1} (r^{(i)}_{L,k-1} - (1-p_{D,k}))(p^{(i)}_{L,k-1}(x|\theta_{k,z})}{1-r^{(i)}_{L,k-1}} \right)^2 \right) \kappa_k(z).
\]  
(30)

Each term of the sum over \( z \) in (30) is non-negative and equals zero only when \( p_{D,k} = 1 \). Hence the bias \( \tilde{N}_k - \bar{N}_k \) is always non-negative, and is zero only when \( p_{D,k} = 1 \). Experimental results in Section V demonstrate that the cardinality bias is significant.

**C. Cardinality Balancing**

Similar to the approach of Mahler in [3], we propose to approximate the PGFI \( U_k[h; z] \) by a Bernoulli \( 1 - r_{U,k}(z) + r_{U,k}(z)p_{U,k}(z; h) \). However, we choose the parameters \( r_{U,k}(z) \) and \( p_{U,k}(z; h) \) so that our proposed Bernoulli approximation has the same intensity function (and hence the same mean cardinality) as that of the original PGFI. More concisely, let \( r_{U,k}(z) \) denote the intensity function of \( U_k[h; z] \), and noting that the intensity function of the Bernoulli is \( r_{U,k}(z)p_{U,k}(z; h) \), we have

\[
r_{U,k}(z)p_{U,k}(z; h) = \nu_{U,k}(z; h).
\]  
(31)

Integrating and normalising (31), respectively, yield the Bernoulli parameters

\[
r_{U,k}(z) = \int \nu_{U,k}(x; z)dx,
\]  
(32)

\[
p_{U,k}(z; h) = \frac{\nu_{U,k}(z; z)}{r_{U,k}(z)}.
\]  
(33)

If \( U_k[h; z] \) is indeed a PGFI of an RFS, then this approach yields the Bernoulli that best approximates \( G_{U,k}[h; z] \) to the first moment.

The intensity function \( \nu_{U,k}(z; h) \) of the PGFI \( G_{U,k}[h; z] \) can be obtained by taking the Frechet derivative\(^4\) of \( G_{U,k}[h; z] \) at \( h = 1 \) in the direction of \( \zeta = \delta_x \) (i.e. the functional derivative at \( x \):)

\[
\nu_{U,k}(x; z) = \frac{\sum_{i=1}^{M_{L,k-1}} (r^{(i)}_{L,k-1}(x|\theta_{k,z})}{\kappa_k(z) + \sum_{i=1}^{M_{L,k-1}} (G_{L,k}^{(i)}[h])}.
\]  
(34)

where

\[
\nu^{(i)}_{L,k}(x; z) = p^{(i)}_{L,k}(x) \left( 1 - r^{(i)}_{L,k-1}(p_{L,k-1}|p_{D,k}) \right)^{-2} \left( \left( 1 - r^{(i)}_{L,k-1}(p_{L,k-1}|p_{D,k}) \right)^{-1} - r^{(i)}_{L,k-1}(p^{(i)}_{L,k}(x)\nu_{L,k}(x; z) - \nu^{(i)}_{L,k}(x; z) \right) \right)
\]  
(35)

In general, \( \nu_{U,k}(x; z) \) is negative whenever \( p_{D,k}(x) = 0 \), and thus, \( p_{U,k}(x; z) \), given by (33), is not a valid probability density. Nonetheless, \( r_{U,k}(z) \), given by (32), agrees with the mean cardinality of \( G_{U,k}[h; z] \) as given by (27). To obtain a valid \( p_{U,k}(z; z) \), we make the approximation \( p_{D,k} \approx 1 \), which eliminates the negative term in (35), i.e.

\[
p_{U,k}(z; z) = \frac{M_{L,k-1} (r^{(i)}_{L,k-1} - (1-p_{L,k}))(p^{(i)}_{L,k-1}(x|\theta_{k,z})}{1-r^{(i)}_{L,k-1}} \right)^2 \right) \kappa_k(z) + \sum_{i=1}^{M_{L,k-1}} (G_{L,k}^{(i)}[h])}
\]  
(36)

Moreover, from the same approximation we have \( p_{U,k}(x; z) \approx 1 \), and consequently (36) reduces to a valid probability density.

The resulting multi-Bernoulli approximation to the updated multi-target density is summarised as follows:\(^5\)

**Proposition 4 (Cardinality-Balanced MeMBer Update).** Under the premises of Proposition 3, if at time \( k \) the predicted multi-target density is a multi-Bernoulli of the form

\[
\pi_{k|k-1} = \left\{ \left( r^{(i)}_{L,k-1}, p^{(i)}_{L,k} \right) \right\}_{i=1}^{M_{L,k-1}}
\]

then the posterior multi-target density can be approximated by a multi-Bernoulli with unbiased cardinality as follows

\[
\pi_k \approx \left\{ \left( \tilde{r}^{(i)}_{U,k}(z), \tilde{p}^{(i)}_{U,k}(z) \right) \right\}_{i=1}^{M_{L,k-1}} \cup \left\{ \left( r^{(i)}_{U,k}(z), p^{(i)}_{U,k}(z; z) \right) \right\}_{z \in Z_k},
\]  
(37)

\(^4\)Defined by \( \lim_{h \to \infty} \frac{G_{U,k}[h+e(z)-G_{U,k}[h]]}{e(z)} \), see [3] pp. 375 eq. (11.186), or [15] pp. 25 eq. (25.1).

\(^5\)The statement of the CBMeMBer update in Proposition 4 is, similar to Proposition 2, also summarized as an approximate update in the interest of readability.
where \( r_{L,k}^{(i)} \), \( p_{L,k}^{(i)} \), \( r_{U,k}^{(i)}(z) \), are given in (14),(15), (27), and
\[
\begin{align*}
p_{U,k}(x; z) &= \frac{\sum_{i=1}^{M_{k|-1}} r_{U,k}^{(i)}(z)}{\sum_{i=1}^{M_{k|-1}} r_{U,k}^{(i)}(z)}/\psi_{k,z}(x). \quad (38)
\end{align*}
\]

Propositions 1 and 4 constitute, respectively, the prediction and update step of the cardinality-balanced MeMBer (CBMeMBer) filter, which propagates the multi-Bernoulli parameters of the posterior multi-target density forward in time.

The complexity of the CBMeMBer recursion is linear in the number of targets and linear in the number of measurements. This is a similar complexity to the PHD filter, but a lower complexity compared to the CPHD filter which is linear in the number of targets and cubic in the number of measurements.

Remark: Note that although the approximation for the posterior PGFI may not be a proper PGFI, it is close to the true posterior PGFI. Hence the approximation given in Proposition 4 is reasonable when clutter is not too dense and the probability of detection is high.

### D. Multi-Target State Estimation

The multi-Bernoulli representation \( \pi_k = \{ (r_k^{(i)}, p_k^{(i)}) \}_{i=1}^{M_k} \) has an intuitive interpretation that facilitates multi-target state estimation from the posterior multi-target density. The existence probability \( r_k^{(i)} \) indicates how likely the \( i \)th hypothesized track is a true track, and the posterior density \( p_k^{(i)} \) describes the estimated current state of the track. Hence, a multi-target state estimate can be obtained by choosing the means or modes from the posterior densities of the hypothesized tracks with existence probabilities exceeding a given threshold (e.g. 0.5). Alternatively, the following basic two-stage procedure can be used. First, we estimate the number of targets from the posterior cardinality distribution by taking its mean or mode (the mode is preferred as it is more stable than the mean). Then, we take the corresponding number of hypothesized tracks having the highest probabilities of existence and compute the individual means or modes from the individual posterior densities.

### E. Extension to Track Propagation

Following the approach in [3], the CBMeMBer recursions can be extended to propagate track continuity, by appropriately labeling and updating the individual Bernoulli components of the posterior multi-Bernoulli density. Thus, given a multi-Bernoulli density \( \pi = \{ (r^{(i)}, p^{(i)}) \}_{i=1}^{M} \), to each (Bernoulli) component \( (r^{(i)}, p^{(i)}) \) we assign a unique (usually integer) track label \( \ell^{(i)} \) to identify hypothesized tracks. The collection of triplets \( T = \{ (\ell^{(i)}, r^{(i)}, p^{(i)}) \}_{i=1}^{M} \) is referred to as a track table. While there are many possible schemes for propagating track labels, a simple scheme is presented as follows.

#### Prediction: If at time \( k - 1 \) the posterior track table is \( T_{k|-1} = \{ (\ell^{(i)}_{k|-1}, r^{(i)}_{k|-1}, p^{(i)}_{k|-1}) \}_{i=1}^{M_{k|-1}} \), then the predicted track table to time \( k \) is
\[
\begin{align*}
T_{k|k-1} &= \{ (\ell^{(i)}_{k|k-1}, r^{(i)}_{k|k-1}, p^{(i)}_{k|k-1}) \}_{i=1}^{M_{k|k-1}} \\
&= \{ (\ell^{(i)}_{k|k-1}, r^{(i)}_{k|k-1}, p^{(i)}_{k|k-1}) \}_{i=1}^{M_{k|-1}} \cup \{ (\ell^{(i)}_{k|k-1}, p^{(i)}_{k|k-1}) \}_{i=1}^{M_{k|k-1}} \\
&= \{ (\ell^{(i)}_{k|k-1}, r^{(i)}_{k|k-1}, p^{(i)}_{k|k-1}) \}_{i=1}^{M_{k|-1}} \cup \{ (\ell^{(i)}_{k|k-1}, \Gamma_{k|k-1}, p^{(i)}_{k|k-1}) \}_{i=1}^{M_{k|k-1}}
\end{align*}
\]
\( \ell^{(i)}_{k|k-1} = \) new label and \( r^{(i)}_{k|k-1} = r^{(i)}_{k|-1}P_{k|k-1}r^{(i)}_{k|k-1}P_{k|k-1}^{(i)} \) are given in Proposition 1. Thus, existing components retain their original labels, and birth components are assigned new labels.

#### Update: If at time \( k \) the predicted track table is \( T_{k|k-1} = \{ (\ell^{(i)}_{k|k-1}, r^{(i)}_{k|k-1}, p^{(i)}_{k|k-1}) \}_{i=1}^{M_{k|k-1}} \), then the updated track table at time \( k \) is \( T_k = \{ (\ell^{(i)}_{k|k-1}, r^{(i)}_{k|k-1}, p^{(i)}_{k|k-1}) \}_{i=1}^{M_{k|-1}} \cup \{ (\ell^{(i)}_{k|k-1}, r^{(i)}_{k|k-1}, P_{U,k}^{(i)}, z) \}_{i=1}^{M_{k|k-1}} \).

In this section, we detail a generic Sequential Monte Carlo (SMC) implementation and an analytic Gaussian mixture (GM) implementation of the proposed CBMeMBer filter.

### A. Sequential Monte Carlo Implementation

In the following, we present a generic SMC implementation of the CBMeMBer recursion which can accommodate non-linear dynamic and measurement models, as well as state-dependent probabilities of survival and detection. This technique directly extends to Maher’s original MeMBer recursion.

#### 1) Prediction: Suppose that at time \( k - 1 \) the (multi-Bernoulli) posterior multi-target density \( \pi_{k-1} = \{ (r_{k-1}^{(i)}, p_{k-1}^{(i)}) \}_{i=1}^{M_{k-1}} \) is given and each \( p_{k-1}^{(i)} \), \( i = 1, \ldots, M_{k-1} \), is comprised of a set of weighted samples \( \{ (\tilde{u}_{k-1}^{(i,j)}, \tilde{x}_{k-1}^{(i,j)}, L_{k-1}^{(i,j)} \} \)
\( \), i.e.
\[
p_{k-1}^{(i)}(x) = \sum_{j=1}^{L_{k-1}^{(i)}} w_{k-1}^{(i,j)} \delta_{\tilde{x}_{k-1}^{(i,j)}}(x).
\]

Then, given importance (or proposal) densities \( q_k^{(i)}(\cdot|\tilde{x}_{k-1}, Z_k) \) such that \( \text{support}(p_k^{(i)}) \subseteq \text{support}(q_k^{(i)}) \) and \( \delta^{(i)}(\cdot|Z_k) \) such that \( \text{support}(p_k^{(i)}) \subseteq \text{support}(\delta^{(i)}) \), the predicted (multi-Bernoulli) multi-target density \( \pi_k = \{ (r_{k|k-1}^{(i)}, p_{k|k-1}^{(i)}) \}_{i=1}^{M_{k|k-1}} \cup \{ (r_{k|k-1}^{(i)}, p_{k|k-1}^{(i)}) \}_{i=1}^{M_{k|k-1}} \)

can be computed as follows
\[
\begin{align*}
r_{k|k-1}^{(i)} &= r_{k-1}^{(i)} \sum_{j=1}^{L_{k-1}^{(i)}} w_{k-1}^{(i,j)} p_{S,k}(\tilde{x}_{k-1}^{(i,j)}), \\
p_{k|k-1}^{(i)}(x) &= \sum_{j=1}^{L_{k-1}^{(i)}} w_{k|k-1}^{(i,j)} \delta_{\tilde{x}_{k}^{(i,j)}}(x), \\
r_{k|k}^{(i)} &= \text{parameter given by birth model}, \\
p_{k|k}^{(i)}(x) &= \sum_{j=1}^{L_{k|k}^{(i)}} w_{k|k}^{(i,j)} \delta_{\tilde{x}_{k}^{(i,j)}}(x)
\end{align*}
\]

where \( x_{k|k}^{(i,j)} \sim q_k^{(i)}(\cdot|\tilde{x}_{k-1}^{(i,j)}, Z_k) \), \( j = 1, \ldots, L_{k|k}^{(i)} \)}}
eliminates particles with low weights and multiplies particles to reduce the effect of degeneracy, we resample the particles for each hypothesis track. To update the number of particles, at each time step pruning of hypothesized tracks is performed by discarding those with existence probabilities below a threshold $P$ (e.g. $10^{-3}$). For the remaining hypothesized tracks, similar to the SMC-PHD/CPHD filters, it is desirable to allocate the number of particles in the track density to be proportional to the expected number of objects present. Thus, at each time step the number of particles given to each hypothesized track is reallocated in proportion to its probability of existence, i.e. during the prediction we sample $L_{1:k} = r_{1:k} L_{\text{max}}$ particles per target, and during resampling we resample $L_{1:k} = r_{k} L_{\text{max}}$ particles for each updated track.

2) Update: Suppose that at time $k$ the predicted (multi-Bernoulli) multi-target density $\pi_{k|k-1} = \{\{r_{i,k|k-1}, p_{i|k|k-1}\}\}_{i=1}^{M_{k|k-1}}$ is given and each $p_{i|k|k-1}$, $i = 1, \ldots, M_{k|k-1}$, is comprised of a set of weighted samples $\{w_{i,k|k-1}, x_{i,k|k-1}\}_{j=1}^{L_{i,k|k-1}}$, i.e.

$$p_{i|k|k-1} = \sum_{j=1}^{L_{i,k|k-1}} w_{i,k|k-1} \delta_{x_{i,k|k-1}}(x).$$

Then, the (multi-Bernoulli) approximation of $\pi_{k|k} = \{\{r_{U,k}(z), p_{U,k}(z)\}\}_{i=1}^{M_{k|k-1}} \cup \{r_{U,k}(z), p_{U,k}(z)\}_{j=1}^{L_{1:k}}$, can be computed as follows:

$$r_{L,k}^{(i)} = \frac{r_{k|k-1}^{(i)} - \varrho_{L,k}^{(i)}}{1 - r_{k|k-1}^{(i)}},$$

$$p_{L,k}^{(i)}(x) = \sum_{j=1}^{L_{i,k|k-1}} w_{i,k|k-1} \delta_{x_{i,k|k-1}}(x),$$

$$r_{L,k}^{*}(z) = \frac{\sum_{i=1}^{M_{i,k|k-1}} \frac{r_{i,k|k-1}^{(i)} - (1 - r_{i,k|k-1}^{(i)})}{1 - r_{i,k|k-1}^{(i)}}}{\varrho_{L,k}^{(i)}},$$

$$p_{L,k}^{*}(x; z) = \sum_{i=1}^{M_{i,k|k-1}} \sum_{j=1}^{L_{i,k|k-1}} w_{i,k|k-1} \delta_{x_{i,k|k-1}}(x),$$

where

$$\varrho_{L,k}^{(i)} = \sum_{j=1}^{L_{i,k|k-1}} w_{i,k|k-1} PD_{k}(x_{i,k|k-1}),$$

$$w_{L,k}^{(i)} = w_{i,k|k-1}^{(i)} / \sum_{j=1}^{L_{i,k|k-1}} w_{i,k|k-1}^{(i)}$$

$$w_{L,k}^{*} = \left(1 - PD_{k}(x_{i,k|k-1})\right),$$

$$\varrho_{L,k}^{*}(z) = \sum_{j=1}^{L_{i,k|k-1}} w_{i,k|k-1} \delta_{x_{i,k|k-1}}(x_{i,k|k-1}),$$

$$w_{U,k}^{*}(x; z) = \sum_{i=1}^{M_{i,k|k-1}} \sum_{j=1}^{L_{i,k|k-1}} w_{i,k|k-1}^{*} \delta_{x_{i,k|k-1}}(x; z),$$

$$w_{U,k}^{*}(x; z) = w_{i,k|k-1}^{*} / \sum_{j=1}^{L_{i,k|k-1}} w_{i,k|k-1}^{*} \delta_{x_{i,k|k-1}}(x; z).$$

3) Resampling and Implementation Issues: Analogous to the standard particle filter, degeneracy is inevitable [14]. To reduce the effect of degeneracy, we resample the particles for each hypothesized track after the update step. This effectively eliminates particles with low weights and multiplies particles with high weights to focus on the important zones of the (single-target) space. Notice that there are many resampling schemes available, and that the choice of resampling scheme affects the computational load as well as the Monte Carlo approximation error [21], [22]. For simplicity however, multinomial resampling is used for the numerical studies in this paper.

Notice that the number of particles required to represent the posterior multi-object density increases due to the birth of objects in the prediction and the averaging of hypothesized tracks in the update. To reduce the number of particles, at each time step pruning of hypothesized tracks is performed by discarding those with existence probabilities below a threshold $P$ (e.g. $10^{-3}$). For the remaining hypothesized tracks, similar to the SMC-PHD/CPHD filters, it is desirable to allocate the number of particles in the track density to be proportional to the expected number of objects present. Thus, at each time step the number of particles given to each hypothesized track is reallocated in proportion to its probability of existence, i.e.

$$\sum_{i=1}^{M_{i,k|k-1}} \frac{r_{i,k|k-1}^{(i)}}{1 - r_{i,k|k-1}^{(i)}}PD_{k}(x_{i,k|k-1}) \approx \sum_{i=1}^{M_{i,k|k-1}} \frac{r_{i,k|k-1}^{(i)}}{1 - r_{i,k|k-1}^{(i)}}\varrho_{L,k}^{(i)},$$

$$p_{L,k}^{*}(x; z) = \sum_{i=1}^{M_{i,k|k-1}} \sum_{j=1}^{L_{i,k|k-1}} w_{i,k|k-1}^{*} \delta_{x_{i,k|k-1}}(x; z),$$

$$w_{U,k}^{*}(x; z) = w_{i,k|k-1}^{*} / \sum_{j=1}^{L_{i,k|k-1}} w_{i,k|k-1}^{*} \delta_{x_{i,k|k-1}}(x; z).$$

B. Closed Form Linear Gaussian Implementation

In the following, we present a closed form solution to the CBMeMBer recursion for the class of linear Gaussian multi-target models which consists of standard linear Gaussian assumptions for the transition and observation models of individual targets, as well as certain assumptions on birth, death and detection:

- Each target follows a linear Gaussian dynamical and observation model i.e.

$$f_{k|k-1}(x|z) = N(x; F_{k-1}z, Q_{k-1}),$$

$$g_{k}(z|x) = N(z; H_{k}x, R_{k}),$$

where $N(\cdot; m, P)$ denotes a Gaussian density with mean $m$ and covariance $P$, $F_{k-1}$ is the state transition matrix, $Q_{k-1}$ is the process noise covariance, $H_{k}$ is the observation matrix, and $R_{k}$ is the observation noise covariance.
The survival and detection probabilities are state independent, i.e.

\[ p_{S,k}(x) = p_{S,k}, \quad p_{D,k}(x) = p_{D,k}. \]  
(49)

The birth model is a multi-Bernoulli with parameter set \( \{ (r_{\Gamma,k}, P_{\Gamma,k}) \}_{k=1}^{M_{\Gamma,k}} \) where \( P_{\Gamma,k} = 1, \ldots, M_{\Gamma,k} \) are Gaussian mixtures of the form

\[ p_{\Gamma,k}(i,k) = \sum_{j=1}^{I_k} w_{\Gamma,k}^{(i,j)} \mathcal{N}(x; m_{\Gamma,k}^{(i,j)}, P_{\Gamma,k}^{(i,j)}), \]  
(51)

with \( w_{\Gamma,k}, m_{\Gamma,k}^{(i,j)}, P_{\Gamma,k}^{(i,j)} \) denoting the weights, means and covariances of the \( j \)-th component.

For the linear Gaussian multi-target model, the following two subsections present a closed form solution to the CB-MeMBer recursion by showing how the posterior density is analytically propagated in time.

1) Prediction: Suppose that at time \( k-1 \) the (multi-Bernoulli) posterior multi-target density \( \pi_{k-1} = \{ (r_{P,k}^{(i,k)}, p_{P,k}^{(i,k)}) \}_{i=1}^{M_{T,k}} \) is given and each probability density \( p_{k-1}^{(i)} \), \( i = 1, \ldots, M_{k-1} \), is comprised of Gaussian mixtures of the form

\[ p_{k-1}^{(i)}(x) = \sum_{j=1}^{I_{k-1}} w_{k-1}^{(i,j)} \mathcal{N}(x; m_{k-1}^{(i,j)}, P_{k-1}^{(i,j)}). \]  
(52)

Then, the predicted (multi-Bernoulli) multi-target density \( \pi_{k-1} = \{ (r_{P,k}^{(i,k)}, p_{P,k}^{(i,k)}) \}_{i=1}^{M_{T,k}} \) can be computed as follows: \( (r_{P,k}^{(i,k)}, p_{P,k}^{(i,k)}) \), \( i, \ldots, M_{T,k} \) are given by the birth model (51), while

\[ r_{P,k}^{(i,k)} = r_{k-1}^{(i)} p_{S,k}, \]  
(53)

\[ p_{P,k}^{(i,k)}(x) = \sum_{j=1}^{I_{k-1}} w_{k-1}^{(i,j)} \mathcal{N}(x; m_{P,k}^{(i,j)}, P_{P,k}^{(i,j)}), \]  
(54)

where

\[ m_{P,k}^{(i,j)} = F_{k-1} m_{k-1}^{(i,j)}, \]  
(55)

\[ P_{P,k}^{(i,j)} = Q_{k-1} + F_{k-1} P_{k-1}^{(i,j)} F_{k-1}^T. \]

2) Update: Suppose that at time \( k \) the predicted (multi-Bernoulli) multi-target density \( \pi_{k-I_{k-1}} = \{ (r_{P,k}^{(i,k)}, p_{P,k}^{(i,k)}) \}_{i=1}^{M_{T,k}} \) is given and each probability density \( p_{k-1}^{(i,k)} \), \( i = 1, \ldots, M_{k-1} \), is comprised of Gaussian mixtures of the form

\[ p_{k-1}^{(i,k)}(x) = \sum_{j=1}^{I_{k-1}} w_{k-1}^{(i,j)} \mathcal{N}(x; m_{k-1}^{(i,j)}, P_{k-1}^{(i,j)}). \]  
(56)

Then, the (multi-Bernoulli) approximation of the updated density \( \pi_k = \{ (r_{U,k}(z), p_{U,k}(z)) \}_{z=1}^{2} \) can be computed as follows

\[ r_{U,k}^{(i)} = r_{k-1}^{(i)} \frac{1 - p_{P,k}^{(i,k)}}{1 - r_{k-1}^{(i)} p_{P,k}^{(i,k)}}, \]  
(57)

\[ p_{U,k}^{(i)}(x) = p_{k-1}^{(i,k)}(x). \]  
(58)

The derivation of the closed form prediction and update steps involves analytically calculating products of Gaussians and integrals of Gaussians using the same approach as in [11], [12]. Full details of this derivation can be found in [15].

Remark: The proposed closed form recursions can easily be extended to accommodate exponential mixture forms for \( p_{S,k} \) and \( p_{P,k} \), following the same line of arguments as for the Gaussian Mixture PHD filter in [11]. Simple approximations such as those proposed in [23], may instead be suitable for accommodating non-constant \( p_{S,k} \) and \( p_{P,k} \) in certain applications. These however, will not be pursued here.

3) Implementation Issues: Notice that the number of Gaussians required to represent the multi-Bernoulli posterior density exactly increases without bound due to the birth of objects in the prediction and averaging of hypothesized tracks in the update. To reduce the number of components, at each time step pruning of hypothesized tracks is performed by discarding those with existence probabilities below a threshold \( P \) (e.g. \( 10^{-3} \)). For each of the remaining tracks, to reduce the number of components comprising its spatial density, we eliminate components with weights below a threshold \( T \), and merge components within a distance \( U \) of each other, analogous to the Gaussian mixture PHD and CPHD filters [11], [12]. In addition it is necessary to impose a maximum of \( J_{\text{max}} \) components per hypothesized track, see [11], [12] for the exact meaning of these parameters.

4) Multi-Target State Estimation: Similar to the SMC implementation, the estimated number of targets can be the cardinality mean or mode, though the latter is preferred. Unlike the SMC implementation, estimates other than the means are possible for individual states. For example, since each posterior density is a Gaussian mixture, it is possible to compute the mode (if the Gaussian components are well-separated this is straightforward). For simplicity we use the
mean of the Gaussian component with the highest weight in our numerical study.

5) Closed Form Linearized and Unscented Implementations: The proposed closed form solution to the CBMeMBer recursion for linear Gaussian models can be extended to accommodate non-linear Gaussian dynamical and observation models with standard approximation methods. Analogous to the linearization and unscented transform strategies of the extended and unscented Kalman filters, we propose the extended and unscented Kalman CBMeMBer filters respectively. Since the proposed extensions are conceptually straightforward, we will not present explicit equations but instead briefly describe the basic approach for the approximate recursions. For the extended-Kalman (EK) version, the closed form expressions for the prediction and update of individual Gaussian components are approximated substituting local linearizations of the non-linear dynamical and observation models respectively. For the unscented-Kalman (UK) version, the prediction and update of individual Gaussian components are approximated by applying the unscented transform to analytically propagate means and covariances through the non-linear dynamical and observation models respectively.

V. NUMERICAL STUDIES

In this section, we demonstrate the performance of the SMC-CBMeMBer and GM-CBMeMBer filters proposed in Section IV with two separate examples and compare with the PHD, CPHD and original MeMBer filters. We evaluate filter performance using the Optimal Sub-Partition Assignment (OSPA) multi-target miss-distance. The miss-distance, or error between the estimated and true state, is fundamental for any given instant with various births and deaths throughout the scenario. The true trajectories are shown in Fig. 1 along with the start and stop positions of each track.

Fig. 1: Target trajectories in the $r\theta$ plane. Start/Stop positions for each track are shown with $O_{t\Delta}$.

The following non-linear dynamical and measurement models are used. A nearly constant turn model having varying turn rate together with noisy bearings and range measurements is considered. The target state variable $x_k = [\tilde{x}_k^T, \omega_k]^T$ comprises the planar position and velocity $\tilde{x}_k^T$ as well as the turn rate $\omega_k$. The state transition model is

$$\begin{align*}
\dot{x}_k &= F(x_{k-1},\omega_{k-1}) + G u_{k-1} \\
\omega_k &= \omega_{k-1} + \Delta u_{k-1}
\end{align*}$$

where

$$F(\omega) = \begin{bmatrix}
\sin \omega & 0 & -1 - \cos \omega \\
\cos \omega & 0 & \sin \omega \\
0 & 1 & \sin \omega \\
0 & 0 & \cos \omega
\end{bmatrix}$$

and

$$u_{k-1} \sim N(0; \sigma_u^2), \quad w_{k-1} \sim N(0; \sigma_d^2 I), \quad \Delta = 1s, \quad \sigma_u = 15m/s^2, \quad \sigma_d = \pi/180 rad/s.$$ 

If detected the observation is a noisy bearing and range vector given by

$$z_k = \left[ \arctan\left(p_{x_k}/p_{y_k}\right) \right] + \varepsilon_k$$

where

$$\varepsilon_k \sim N(0; R_k), \quad R_k = \text{diag}(\sigma_u^2, \sigma_d^2), \quad \sigma_u = (\pi/180)rad, \quad \sigma_d = 5m.$$
At each time step in the SMC implementations of the CBMeMber and MeMber filters, a maximum of $L_{\text{max}} = 1000$ and minimum of $L_{\text{min}} = 300$ particles per hypothesized track is imposed, with resampling performed so that the number of particles representing each track is proportional to its probability of existence. Additionally, pruning of hypothesized tracks is performed with a weight threshold of $P = 10^{-3}$ and a maximum of $T_{\text{max}} = 100$ tracks. Fig. 2 plots the $x$ and $y$ components (versus time) of the measurements, true trajectories, and SMC-CBMeMber filter estimates. The plots indicate that the SMC-CBMeMber filter is able to identify all target births and deaths, as well as successfully accommodating nonlinearities. The filter has no difficulty handling target crossings, in particular the crossing of three targets at time $k = 50$, and a another pair at time $k = 80$.

![Fig. 2. Measurements, true tracks, and SMC-CBMeMber filter estimates.](image)

1) Monte Carlo Verification: To evaluate the performance of the SMC-CBMeMber filter, we compare it with the SMC-PHD, SMC-CPHD and SMC-MeMber filters over 1000 Monte Carlo (MC) trials. At each trial, the same target tracks shown in Fig. 1 are used but a new set of measurement data is randomly generated. Fig. 3(a)-(d) shows the mean and standard deviation of the estimated cardinality distribution versus time for the filters under study. These results confirm that the CBMeMber, PHD and CPHD filters are unbiased in estimating the number of targets, whereas the MeMber filter has a significant positive bias. In addition, the CBMeMber filter has a lower variance on its estimated cardinality than the PHD filter, but has a larger variance than the CPHD filter. This observation can be attributed to the fact that the CBMeMber filter propagates a parameterized approximation to the posterior cardinality distribution, whereas the PHD filter propagates only the corresponding mean and the CPHD filter propagates the entire distribution.

In Fig. 4, the MC averages of the OSPA distance (for $p = 1$ and $c = 300$) are shown versus time. It can be seen that the SMC-CBMeMber, SMC-CPHD, SMC-PHD and SMC-MeMber filters produce average errors of approximately 50m, 60m, 70m, and 160m per-target. These results suggest that the CBMeMber filter outperforms the PHD filter. This is due to the previously mentioned drawbacks in the way that the latter extract state estimates. Note also that the error in the CPHD filter is noticeably lower than in the PHD filter due to the difference in the variance of their cardinality estimates observed in Fig. 3. It can also be seen that the MeMber filter performs significantly worse than the other filters under consideration. The large penalty observed here can be attributed to the large cardinality errors produced by this filter as seen in Fig. 3.

2) Testing of Limitations: We now briefly discuss the extent of the limitations of the proposed cardinality-balancing. Since the CBMeMber filter is derived under the assumption of reasonably low clutter, we investigate the breakdown of the proposed filter by increasing the clutter rate to $\lambda_c = 8.0 \times 10^{-3} (\text{radm})^{-1}$ (giving an average of 50 clutter returns per scan). 1000 MC trials are again performed for our SMC implementations. In Fig. 5, the mean and standard deviation of the estimated cardinality distribution versus time are shown for all filters under consideration. It can be seen that the CBMeMber filter exhibits a noticeable bias, although the extent of the bias is still significantly less than the MeMber filter. In contrast, the CPHD filter remains unbiased in such conditions, whereas the PHD filter exhibits noticeable signs of breakdown. Empirically, our experiments have suggested that the proposed SMC-CBMeMber filter performs well up to a clutter intensity of approximately $\lambda_c = 3.20 \times 10^{-3} (\text{radm})^{-1}$.

![Fig. 3. Cardinality statistics for the (a) SMC-CBMeMber filter (b) SMC-PHD filter (c) SMC-CPHD filter (d) SMC-MeMber filter](image)

![Fig. 4. OSPA distances for the SMC-CBMeMber, SMC-PHD, SMC-CPHD and SMC-MeMber filters](image)
(giving an average of 20 clutter returns per scan). Above this value, the bias in the filter’s cardinality estimate becomes increasingly noticeable.

![Cardinality statistics for a high clutter rate for (a) SMC-CBMeMBer filter (b) SMC-PHD filter (c) SMC-CPHD filter (d) SMC-MeMBer filter.](image)

Fig. 5. Cardinality statistics for a high clutter rate for (a) SMC-CBMeMBer filter (b) SMC-PHD filter (c) SMC-CPHD filter (d) SMC-MeMBer filter

3) Overall Evaluation: In Fig. 6, the 1000 MC trial averages of the OSPA distance (time-averaged over the duration of the scenario) for the SMC-CBMeMBer, SMC-PHD, SMC-CPHD and SMC-MeMBer filters are shown against clutter intensities from \( \lambda_c = 0 \text{ (radm)}^{-1} \) to \( \lambda_c = 8 \times 10^{-3} \text{ (radm)}^{-1} \). As expected, the miss-distances increase with higher clutter intensities. Moreover, it appears that the SMC-CBMeMBer filter outperforms the SMC-CPHD filter which in turn outperforms the SMC-PHD filter (with the latter exhibiting breakdown at higher clutter), whereas the SMC-MeMBer filter performs poorly overall (primarily due to cardinality errors). Note that the result of time-averaging should only be viewed as a broad indication of filter performance, since the average is likely to be scenario dependent. The performance over a range of probability of detection values are not considered since \( p_{D,k} (\cdot) \) is state dependent in this scenario.

![Time averaged OSPA distances for varying clutter intensity](image)

Fig. 6. Time averaged OSPA distances for varying clutter intensity

B. Linear Gaussian Example

We demonstrate the performance of the GM-CBMeMBer filter and compare it with the GM-MeMBer, GM-PHD and GM-CPHD filters via the following example. A time varying number of targets is observed in clutter on a two dimensional surveillance region with dimensions \([-1000,1000] \times [-1000,1000] \text{m} \). A maximum of 10 targets appears on the scene at any one time, and target births and deaths occur at various times and locations. The trajectories for each target are shown in Fig. 7.

![Target trajectories in the x-y plane. Start/Stop positions for each track are shown with OΔ.](image)

Fig. 7. Target trajectories in the x-y plane. Start/Stop positions for each track are shown with OΔ.

The following dynamical and measurement models are used. The target state variable is a vector of planar position and velocity:

\[
\begin{bmatrix}
  x_k \\
  y_k \\
  \dot{x}_k \\
  \dot{y}_k
\end{bmatrix} =
\begin{bmatrix}
  x_k + \nu_k \\
  y_k + \nu_k \\
  \dot{x}_k + \nu_k \\
  \dot{y}_k + \nu_k
\end{bmatrix},
\]

where \( \nu_k \) denotes the measurement noise. Clutter is Poisson with intensity \( \lambda_c \text{ (radm)}^{-1} \).

At each time step in the GM-CBMeMBer and GM-MeMBer filters, pruning and merging of Gaussian components are performed for each hypothesized track using a weight threshold of \( T = 10^{-5} \), a merging threshold of \( U = 4 \text{m} \), and a maximum of \( J_{max} = 100 \) components. Additionally, pruning of hypothesized tracks is performed within a weight threshold of \( P = 10^{-3} \) and a maximum of \( T_{max} = 100 \) tracks. In the GM-PHD, GM-CPHD filters used for comparison, the same pruning and merging of Gaussian components is performed on the posterior intensity.
In Fig. 8, the $x$ and $y$ components of the measurements, true trajectories and state estimates from the GM-CBMeMBer filter are shown versus time. These results suggest that the proposed filter is able to correctly track the individual target motions and identify the various target births and deaths throughout. The filter also has no difficulty handling crossings of three targets at time $k = 40$, and two targets at time $k = 60$.

![Fig. 8. Measurements, true tracks, and GM-CBMeMBer filter estimates.](image)

1) Monte Carlo Verification: We verify the performance of the GM-CBMeMBer filter over 1000 Monte Carlo (MC) trials with the one fixed set of target tracks but randomly generated measurement data. In Fig. 9(a)-(d), the mean and standard deviation of the estimated cardinality distribution versus time are shown for the GM-CBMeMBer, GM-PHD, GM-CPHD and GM-MeMBer filters. Similar to the SMC implementations, inspection of these results verifies that the CBMeMBer, PHD and CPHD filters are unbiased in their estimates of the cardinality, whereas the MeMBer filter is significantly biased. Again, these results suggest that the CBMeMBer filter has a lower variance on its estimated cardinality than the PHD filter, but has a larger variance than the CPHD filter.

In Fig. 10, the MC averages of the OSPA distance (for $p = 1$ and $c = 300$) are shown for all filters under consideration. It can be seen that the GM-CPHD, GM-CBMeMBer, GM-PHD and GM-MeMBer filters settle to errors of approximately $17m$, $23m$, $23m$ and $125m$ per-target. In contrast to the SMC case, the Gaussian mixture implementations show that the CBMeMBer filter performs similarly to the PHD filter, but worse than the CPHD filter. This further consolidates the observation that the better multi-target state estimation in the SMC implementation contributes to better performance of the CBMeMBer filter (relative to the PHD, CPHD filters). The disparity with CPHD filter can be expected due to the difference in the variance of their cardinality estimates. However, the similar performance with the PHD filter, despite the difference in the variance of their cardinality estimates, is most likely due to the high SNR in this scenario giving the PHD filter more of an advantage than usual. Again, the poor performance of the original MeMBer filter can be expected as a direct result of its significant cardinality bias.

2) Testing of Limitations: We now briefly demonstrate the breakdown of the proposed cardinality-balancing by sufficiently increasing the clutter rate to $\lambda_c = 1.25 \times 10^{-5}m^{-2}$ (giving an average of 50 clutter returns per scan). Again, 1000 MC trials are performed for the GM implementations. In Fig. 11, the mean and standard deviation of the estimated cardinality distribution versus time are shown for all filters. The CBMeMBer filter now exhibits a noticeable bias which is still significantly smaller than the MeMBer filter, but it is clear that the PHD and CPHD filters remain unbiased in these conditions. Empirically, the proposed GM-CBMeMBer filter performs well up to a clutter intensity of approximately $\lambda_c = 5.00 \times 10^{-6}m^{-2}$, and down to a probability of detection of approximately $p_{D,k} = 0.90$.

3) Overall Evaluation: In Fig. 12, the 1000 MC trial averages of the OSPA distance (time-averaged over the duration of the scenario) for the GM-CBMeMBer, GM-PHD, GM-CPHD and GM-MeMBer filters are shown for various clutter intensities from $\lambda_c = 0m^{-2}$ to $\lambda_c = 1.25 \times 10^{-5}m^{-2}$. As expected, the miss-distances increase with higher clutter intensities. On the whole, these results suggest that the GM-CBMeMBer filter has performance consistent with that of the GM-PHD and GM-CPHD filters, whereas the GM-MeMBer filter performs poorly overall (again due to cardinality errors).

4) EK/UK Approximations: We briefly demonstrate the proposed EK and UK CBMeMBer approximations using the
Mean & StDev

Cardinality Statistics

True Mean StDev

(a) GM−CBMeMBer

(b) GM−PHD

(c) GM−CPHD

(d) GM−MeMBer

Fig. 11. Cardinality statistics for a high clutter rate for (a) CBMeMBer filter

(b) PHD filter (c) CPHD filter (d) MeMBer filter

Fig. 12. Time averaged OSPA distances for varying clutter intensity

VI. CONCLUSION

It has been shown analytically that Mahler’s MeMBer filter has a bias in the number of targets, which vanishes only for unity probability of detection. A new filter, the CBMeMBer filter, which eliminates the posterior cardinality bias has been proposed along with Simultaneous Monte Carlo and Gaussian mixture implementations. The CBMeMBer filter has smaller complexity than the CPHD filter and similar complexity to the PHD filter. Experiments with linear and nonlinear scenarios confirm that the proposed filter drastically reduces the cardinality bias. Under low clutter and high probability of detection, empirical results suggest that, in SMC implementations the CBMeMBer filter outperforms the CPHD and PHD filters, whereas in Gaussian mixture implementations the CPHD filter is still superior and the CBMeMBer can only achieve similar performance to the PHD filter. Thus, for applications where Gaussian mixture implementations are not applicable, e.g.

REFERENCES


