Multitarget multisensor tracking

Santosh Nannuru

Department of Electrical & Computer Engineering
McGill University
Montreal, Canada

August 2015

A thesis submitted to McGill University in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

2015/08/12

Abstract

In this thesis we develop various multitarget tracking algorithms that can process measurements from single or multiple sensors. The filters are derived by approximate application of the recursive Bayes filter within the random finite set framework, which is used to model the multitarget state and observations. The contributions of the thesis can be organized into three main categories.

To provide a motivating application for the algorithms we develop, we first study the problem of radio frequency tomography. We empirically validate a radio frequency tomography measurement model when multiple targets are present within the sensor network. We validate models for both indoor and outdoor environments. These models are then used to perform multitarget tracking using various Monte Carlo filters on data gathered from field deployments of radio frequency sensor networks.

Second, we develop auxiliary particle filter implementations of the Probability Hypothesis Density filter and Cardinalized Probability Hypothesis Density filter when the measurement model has a specific form, namely the superpositional sensor model. We also derive Multi-Bernoulli filter and Hybrid Multi-Bernoulli Cardinalized Probability Hypothesis Density filter for superpositional sensors and develop their auxiliary particle filter implementations. These filters are evaluated for multitarget tracking using simulated radio frequency tomography and acoustic sensor network models.

Third, we derive update equations for the General Multisensor Cardinalized Probability Hypothesis Density filter when the measurement model has a specific form, namely the standard sensor model. To overcome the combinatorial computational complexity of this filter we develop a Gaussian mixture model-based greedy algorithm to implement the filter in a computationally tractable manner. The filter is evaluated using simulated multisensor measurements.

Sommaire

Dans cette thèse nous développons différents algorithmes de pistage multicible qui peuvent traiter les mesures d'un ou plusieurs capteurs. Les filtres sont obtenus par application approximative du filtre de Bayes récursif dans le contexte d'ensemble fini aléatoire, contexte qui est utilisé pour modélisé les états et les observations. Les contributions de la thèse peuvent être organisées en trois parties.

Pour fournir une application motivante des algorithmes que nous développons, nous étudions d'abord le problème de tomographie à radiofréquence. Nous validons empiriquement un modèle de mesure pour tomographie à radiofréquence lorsque plusieurs cibles sont présentes à l'intérieur du réseau de capteurs. Nous validons des modèles pour des environnements à la fois intérieurs et extérieurs. Ces modèles sont ensuite utilisés pour réaliser du pistage multicible utilisant différents filtres de Monte Carlo sur les données capturées lors de déploiements sur le terrain de réseaux de capteurs sans-fil.

En second lieu, nous développons des implémentations de filtres particulaires auxiliaires pour le filtre "Probability Hypothesis Density" et le filtre "Cardinalized Probability Hypothesis Density" lorsque le modèle de mesure possède une forme particulière, à savoir le modèle superposé de capteur. Nous obtenons aussi un filtre multi-Bernouilli et un filtre "Hybrid Multi-Bernoulli Cardinalized Probability Hypothesis Density" pour les capteurs de modèle superposé et développons leurs implémentations de filtres particulaires auxiliaires. Ces filtres sont évalués à des fins de pistage multicible en utilisant de la tomographie à radiofréquence simulée et plusieurs modèles acoustiques de réseaux de capteurs.

En troisième lieu, nous dérivons des équations de mise-à-jour pour le filtre "General Multisensor Cardinalized Probability Hypothesis Density" lorsque le modèle de mesure possède une forme particulière, à savoir le modèle standard de capteur. Pour surmonter la complexité combinatoire de ce filtre, nous développons un algorithme glouton avec mélange de gaussiennes qui est effectivement traitable en temps fini. Le filtre est évalué en utilisant des mesures simulées provenant de différents capteurs.

Acknowledgments

I would like to express my sincere gratitude and thanks to my supervisor Prof. Mark Coates. This thesis would not have been possible without his constant encouragement and guidance. I am extremely thankful for the ample research collaboration opportunities and financial support he has provided me over the course of my doctoral studies. I would also like to express my thanks to Prof. Michael Rabbat for his constructive feedback, and general approachability on numerous occasions.

I had the good fortune of collaborating with many talented researchers. I would like to thank Frederic Thouin, Yunpeng Li, Yan Zeng, Bo Yang, Ronald Mahler, Michael Rabbat and Stephane Blouin for collaborating on various parts of this thesis. I am also grateful to Arnaud Doucet for the technical discussions and guidance he provided during my internship at Oxford university.

My colleagues in the Computer networks lab have always been welcoming and friendly. I would like to thank Yunpeng, Zhe, Deniz, Milad, Tao, Oscar, Arslan, Jun Ye, Shohreh, Guillaume and Syamantak for their companionship over the years. Thanks to Guillaume for translating the abstract of this thesis into french. Special thanks to Rizwan, Alok and Sankha for their friendship, and for making me feel at home.

I am very grateful to my parents, brother and sister for their caring love, support and understanding during the many ups and downs of research.

Finally I would like to thank my dear friend Adi for being with me through all the good and bad times. I would like to dedicate this thesis to the memory of my friend Theja and my grandfathers.

Contents

List of Figures

List of Tables

List of Acronyms

RF	Radio Frequency
RFS	Random Finite Set
PHD	Probability Hypothesis Density
CPHD	Cardinalized Probability Hypothesis Density
PGF	Probability Generating Function
PGFL	Probability Generating Functional
IIDC	Independent and Identically Distributed Cluster
OSPA	Optimal Subpattern Assignment
MCMC	Markov Chain Monte Carlo
GM	Gaussian Mixture
RSS	Received Signal Strength
SIR	Sequential Importance Resampling
MPF	Multiple Particle Filter

Chapter 1

Introduction

Recent advances in sensing capabilities and electronics have led to variety of cheap, easy-to-deploy sensors which can gather data about a wide range of quantities. However, data gathered by these sensors is often imperfect and corrupted by noise. Efficiently extracting useful information from noisy sensor network data coming from multiple sources, in a timely manner, requires non-trivial processing techniques. Bayesian inference is a principled approach to processing such data. But the Bayesian inference process is often computationally challenging. For practical feasibility, we require efficient, computationally tractable and noise-robust algorithms to process sensor network data.

The goal of this thesis is to advance the state-of-the-art multitarget tracking algorithms for processing multisensor data. Towards this goal we utilize the recently developed random finite set framework which provides a mathematically elegant representation of system variables and a unified approach for state estimation.

1.1 Motivation

1.1.1 Radio frequency tomography

Radio-frequency (RF) tomography is the process of monitoring an area to detect mobile targets based on the additional attenuation and fluctuations they cause in wireless transmissions [5]. RF tomography systems have several desirable aspects. Wireless networks of radio-frequency sensors can be easily deployed and are relatively inexpensive. Compared to the other available alternatives such as infrared and video, RF measurements have the

advantage that they can penetrate walls and other non-metallic obstacles. RF sensor networks can be used for military surveillance, search-and-rescue operations, through-the-wall imaging and in healthcare environments.

In previous work, tracking using radio-frequency tomography principles has been primarily limited to a single target [5–8]. Most practical applications where radio-frequency tomography is used require tracking of multiple targets, and the number of targets is possibly unknown and time varying. The existing RF tomography systems have limitations such as restricted sensor network deployment options, extensive training to learn model parameters, and difficulty in tracking multiple targets. To address these issues, in this thesis we develop RF tomography measurement models for multiple targets and validate them by collecting experimental sensor network data. The sensor networks we study can be deployed in both outdoor and indoor locations. The measurement model parameters can be learned using a small amount of training data and are relatively robust across different locations. We also implement various Monte Carlo filters that can perform multitarget tracking in real-time.

1.1.2 Random finite set representation

Traditionally in the multitarget tracking literature, the unknown multitarget states and the observed sensor measurements are modelled as realisations of random vectors. This random vector representation is inefficient when the number of targets and/or the number of measurements are unknown and change over time. A naive solution to this problem is to process vectors of much higher dimensions than required but it comes at the cost of increased algorithmic and computational complexity.

An elegant solution to this problem is the *random finite set* (RFS) representation [9] which models the multitarget states and sensor measurements as realisations of random finite sets. This modelling inherently allows uncertainty in the number of elements of the set and the specific values they take. Several statistical filters can be derived based on the random finite set representation. These operate in the single target state space and are computationally efficient. In this thesis, we develop and implement several multitarget tracking filters for different measurement models using the RFS representation.

1.1.3 Filters for superpositional sensors

The measurements gathered by radio-frequency tomographic tracking systems [4], acoustic sensor networks [10], and antenna arrays [11–13] have a specific likelihood form. We refer to this specific likelihood model as the *superpositional sensor model*. Superpositional sensors have the property that the contribution to a measurement due to multiple targets is equal to the sum of contributions to that measurement from each of the targets when present alone. The contributions of individual targets can potentially be a nonlinear function of the target state. Many of the existing approaches that process superpositional sensors transform the measurements into alternate forms which can lead to loss of information. To accurately process these measurements we need to develop algorithms that can incorporate the specific observation model of superpositional sensors.

The random finite set based filters such as the *probability hypothesis density* (PHD) filter and the *cardinalized probability hypothesis density* (CPHD) filter [14] have been derived for superpositional sensors but they are not practical because of their analytical intractability. Closed form Gaussian mixture model-based implementations of these filters have been proposed [15] but they suffer from high computational complexity and are limited to linear-Gaussian models. In this thesis we derive approximate but computationally tractable random finite set based filters that take into account the specific observation model of superpositional sensors. We also develop particle filter implementations of these filters and use them to perform multitarget tracking using simulated acoustic sensor networks and radio-frequency sensor networks.

1.1.4 Filters for standard sensors

Another measurement model which has been frequently studied in the literature is the *standard sensor model*. Measurements gathered by radar systems [16], sonar systems [17], and range-and-bearing sensors [18] can be modelled using the standard sensor model. In contrast to the superpositional sensor model, a sensor following the standard sensor model gathers multiple measurements where each measurement is either associated with a target or is a clutter measurement. Though a significant amount of multitarget tracking research based on random finite set representation has focused on the standard sensor model, most of it is limited to the case where a single sensor is making observations. Given the widespread presence of sensor networks where using multiple sensors (possibly unreliable and of low-

observability) can potentially provide more accurate tracking information, there is a need for robust algorithms that can process these sensor measurements in a holistic manner.

Several algorithms have been proposed in the literature to process multisensor measurements [19–21] but they have drawbacks restricting their applicability. The drawbacks include *(i)* sensor order dependence - the tracking performance depends on the order in which sensors are processed and the performance can severely deteriorate if unreliable sensors are processed last; *(ii)* numerical instability; and *(iii)* computational intractability - the multisensor multitarget tracking problem is inherently combinatorial in nature and efficient filter implementations are required for practical usability. To address these issues, in this thesis we derive a general multisensor CPHD filter for standard sensors. To overcome its combinatorial complexity we propose a greedy implementation which *(i)* has significantly reduced dependence on the order in which sensors are processed; *(ii)* has a numerically stable implementation; and *(iii)* is computationally feasible with complexity increasing linearly with the number of sensors.

1.2 Thesis organization and contributions

We now detail the organization of the thesis into chapters and summarize the major technical contributions:

- In *Chapter 2* we provide background material required for the thesis. Specifically, we state the multitarget tracking problem statement and provide a brief discussion of the optimal multitarget Bayes filter. We introduce the concept of random finite sets, their related statistics, and some common examples of random finite sets. We also discuss the standard sensor model and the superpositional sensor model. An extensive literature review discussing the state-of-the-art multitarget multisensor tracking algorithms is also provided.

- In *Chapter 3* we propose and validate empirical measurement models when multiple targets are present for the radio-frequency passive localization sensor network. Models are validated for both indoor and outdoor environments. We collect measurements when multiple targets are present within the radio frequency sensor network and process them offline to successfully demonstrate tracking of up to four targets in outdoor environments and up to three targets in indoor environments.

- In *Chapter 4* we propose auxiliary particle filter based implementations of the PHD and CPHD filters for superpositional sensors. These filter implementations are used to estimate the number and location of targets in a simulated radio frequency sensor network and a simulated acoustic sensor network.

- In *Chapter 5* we derive update equations for the approximate multi-Bernoulli filter and approximate hybrid multi-Bernoulli CPHD filter for superpositional sensors. We also propose auxiliary particle filter based implementations of these approximate filters. Successful tracking of multiple time varying number of targets is demonstrated using simulated radio frequency sensor network measurements and simulated acoustic sensor network measurements.

- In *Chapter 6* we derive update equations for the general multisensor CPHD filter for standard sensors. Since the exact update equations have combinatorial computational complexity, we propose a greedy algorithm for implementing the general multisensor PHD and CPHD filters using Gaussian mixture models. We use simulated data to verify the feasibility of the proposed algorithms and compare them with other approximate multisensor algorithms. An unscented Kalman filter based implementation of the greedy algorithm is also proposed and verified using simulated data.

- In *Chapter 7* we provide conclusions of this thesis. We also identify future research directions to continue and improve on the algorithms developed in this thesis.

Published work

The following publications have resulted from the research work reported in this thesis. The work has been conducted in collaboration with Prof. Mark Coates, Frederic Thouin, Yunpeng Li, Yan Zeng, Prof. Bo Yang, Ronald Mahler, Prof. Michael Rabbat and Stephane Blouin.

- *Chapter 3*

  - ⋆ S. Nannuru, Y. Li, Y. Zeng, M. Coates, and B. Yang, "Radio frequency tomography for passive indoor multi-target tracking", *IEEE Transactions on Mobile Computing.*, vol. 12, pp. 2322-2333, Dec. 2013 [22].

Experimental data collection was done by Yunpeng Li, Yan Zeng and I with the help of other students at McGill University and by students at Beijing University of Posts and Telecommunications under the direction of Prof. Bo Yang. Yunpeng Li and Yan Zeng are additionally responsible for verification of measurement models using single link data and single target tracking. My role was to implement different multitarget tracking algorithms and analyze them using the collected field data.

⋆ S. Nannuru, Y. Li, M. Coates, and B. Yang, "Multi-target device-free tracking using radio frequency tomography", in *Proc. Int. Conf. Intelligent Sensors, Sensor Networks and Information Processing*, Adelaide, Australia, Dec. 2011 [23].

Yunpeng Li is responsible for measurement model verification using single link data, providing help with experimental data collection at McGill University and single target tracking. Prof. Bo Yang directed students at Beijing University of Posts and Telecommunications for the data collection process. My role was to implement different multitarget tracking algorithms and analyze them using the collected field data.

- *Chapter 4*

⋆ F. Thouin, S. Nannuru, and M. Coates, "Multi-target tracking for measurement models with additive contributions", in *Proc. Int. Conf. Information Fusion*, Chicago, IL, U.S.A., July 2011 [4].

Prof. Mark Coates derived the approximate PHD filter equations. Frederic Thouin is responsible for developing its particle implementation. My role was to implement the MCMC filter for comparison with the proposed PHD filter.

⋆ S. Nannuru, M. Coates, and R. Mahler, "Computationally-tractable approximate PHD and CPHD filters for superpositional sensors", *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, pp. 410-420, Jun. 2013 [24].

Ronald Mahler is responsible for derivation of the CPHD filter equations. I am responsible for developing particle implementations of the different filters and performing numerical studies.

- *Chapter 5*
  - ⋆ S. Nannuru and M. Coates, "Multi-Bernoulli filter for superpositional sensors", in *Proc. Int. Conf. Information Fusion*, Istanbul, Turkey, Jul. 2013 [25].
  - ⋆ S. Nannuru and M. Coates, "Particle filter implementation of the multi-Bernoulli filter for superpositional sensors", in *Proc. Int. Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, Saint Martin, Dec. 2013 [26].
  - ⋆ S. Nannuru and M. Coates, "Hybrid multi-Bernoulli CPHD filter for superpositional sensors", *Proc. SPIE Int. Conf. Signal Processing, Sensor Fusion and Target Recognition*, Baltimore, MD, U.S.A., May 2014 [27].
  - ⋆ S. Nannuru and M. Coates, "Hybrid multi-Bernoulli and CPHD filters for superpositional sensors", accepted in *IEEE Transactions on Aerospace and Electronic Systems* [28] (14 pages).
- *Chapter 6*
  - ⋆ S. Nannuru, M. Coates, M. Rabbat, and S. Blouin, "General solution and approximate implementation of the multisensor multitarget CPHD filter", Proc. Int. Conf. Acoustics, Speech and Signal Processing, Brisbane, Australia, Apr. 2015 [29].

    Prof. Michael Rabbat helped in developing the approximate filter implementation and provided technical feedback on numerical simulations. Stephane Blouin provided editorial feedback on the manuscript. I am responsible for developing approximate implementation of the filter and performing numerical simulations.

The following submission is currently under the review process.

- S. Nannuru, S. Blouin, M. Coates, and M. Rabbat, "Multisensor CPHD filter", submitted to *IEEE Transactions on Aerospace and Electronic Systems* [30] (17 pages, submission reference number: TAES-201500265, date of submission: April 13, 2015).

  The majority of the content of *Chapter 6* is based on this paper. Stephane Blouin is responsible for providing the sea-trial data set and for editorial feedback on the manuscript. Prof. Michael Rabbat helped in developing the approximate filter implementation and provided technical feedback on derivations and numerical simulations.

I am responsible for deriving filter equations, developing approximate filter implementation, and performing numerical studies.

1.3 Scope of the thesis

The problem of multitarget multisensor tracking encompasses a broad range of topics and techniques. In this section we will briefly discuss the scope of this thesis. The multisensor multitarget tracking system under investigation is a centralized system. We assume that observations from all the sensors are aggregated at a central computer which performs the task of inference. This should be contrasted with decentralized tracking systems where the individual nodes are capable of both collecting and processing the data.

The radio frequency tomography measurement models we develop are empirical models. Study of radio frequency wave propagation and modelling using principles of physics is beyond the scope of this thesis.

A significant amount of past literature has focused on multitarget tracking techniques which simultaneously solve the problem of tracking and data association. Data association refers to the problem of associating targets with measurements when standard sensors are used. Examples of such filters are the joint probabilistic data association (JPDA) filter [31] and the multiple hypothesis tracking (MHT) filter [32]. In this thesis we do not explicitly consider the problem of data association and the focus is on developing algorithms which do not require data association step.

In the thesis we consider linear-Gaussian target dynamics. We assume that each target evolves independently, follows a Markov evolution model and does not interact with other targets. Birth and death of targets is allowed but target spawning is not considered. Note that though we restrict our study to linear-Gaussian target dynamics, several of the filters we develop can be easily extended to account for non-linear non-Gaussian dynamics. Both linear and non-linear sensor observation models are considered. For the standard sensors we consider set-valued clutter noise and additive Gaussian noise when targets are detected. For superpositional sensors we consider additive Gaussian noise.

The multitarget tracking algorithms studied and analyzed fall under the Bayesian filtering paradigm. The knowledge about the system models is fused with the incoming observations to update the multitarget state information using Bayes rule. The optimal recursive Bayes filter in its original form is mathematically intractable in most cases of

interest. Various parametric and non-parametric approximations of the Bayes filter are considered. Filter implementations are constructed using Monte Carlo approximations or Gaussian mixture model-based approximations.

Chapter 2

Background and literature review

In this chapter we provide a brief background on the various concepts required in this thesis. A detailed literature survey of the state-of-the-art in the multitarget multisensor tracking research is also included. The organization of the chapter is as follows. In Section 2.1 we describe the multitarget multisensor problem statement that we address in this thesis. An introduction to random finite sets is given in Section 2.2. Examples of random finite sets are given in Section 2.3. The superpositional sensor model and the standard sensor model are discussed in Section 2.4. In Section 2.5 we provide prediction and update equations for the recursive Bayes filter. In Section 2.6 we review some properties of Gaussian distributions that will be used in this thesis. The metric used to compare different tracking algorithms is provided in Section 2.7. Detailed literature reviews on radio-frequency tomography, filters for standard sensors, and filters for superpositional sensors are provided in Sections 2.8, 2.9, and 2.10 respectively.

2.1 Multitarget tracking problem statement

We now specify the multisensor multitarget tracking problem. Let $\mathbf{x}_{k,i} \in \mathcal{X}$ be the state of the i^{th} target at time k. In most of the tracking literature $\mathcal{X}$ is chosen to be the Euclidean space, $\mathcal{X} = \mathbb{R}^{n_{\mathbf{x}}}$, where $n_{\mathbf{x}}$ is the dimension of the single target state. If $n_k \geq 0$ targets are present at time k, the multitarget state can be either represented using a vector $\mathbf{X}_k = [\mathbf{x}_{k,1}, \ldots \mathbf{x}_{k,n_k}]$ or represented by a finite set $X_k = \{\mathbf{x}_{k,1}, \ldots \mathbf{x}_{k,n_k}\}$, $X_k \subseteq \mathcal{X}$. In this thesis we mostly use the finite set representation and develop filters based on this representation. We assume that each single target state evolves according to the Markovian transition function

$t_{k+1|k}(\mathbf{x}_{k+1,i}|\mathbf{x}_{k,i})$. New targets can arrive and existing targets can disappear at each time step. The survival probability of an existing target with state $\mathbf{x}$ at time k is denoted by the function $p_{sv,k}(\mathbf{x})$.

Multiple sensors make observations about the multiple targets present within the monitored region. We discuss the specific form of the measurement model in later sections of this chapter. Let the measurement at time step k be represented by the vector $\mathbf{Z}_k$ or the set Z_k. The objective of the multitarget tracking problem is to form an estimate $\widehat{\mathbf{X}}_k$ or $\widehat{X}_k$ of the multitarget state at each time step k. This estimate is formed using all the measurements up until time k which is denoted by $Z_{1:k} = \{Z_1, Z_2, \ldots, Z_k\}$ or $\mathbf{Z}_{1:k} = \{\mathbf{Z}_1, \mathbf{Z}_2, \ldots, \mathbf{Z}_k\}$. More generally, we would like to estimate the posterior multitarget state distribution $f_{k|k}(X_k \text{ or } \mathbf{X}_k|Z_{1:k} \text{ or } \mathbf{Z}_{1:k})$.

2.2 Random finite sets

Random sets are an extension of the concept of random variables and random vectors. While random vectors are of a predefined dimension and have an ordering of their elements, random sets can have uncertainty in the set dimension and there is no preferred ordering of the elements of the set. We now give a very brief formal introduction to random finite sets. For a detailed mathematical treatment of random finite sets readers are advised to refer to the literature [33–36].

Let $(\Omega, \sigma(\Omega), P)$ be a probability space where Ω is the sample space, $\sigma(\Omega)$ is the σ-algebra of subsets of Ω, and P is a measure defined over $\sigma(\Omega)$. Let E be a locally compact Hausdorff separable space and let $\mathcal{F}(E)$ be the collection of finite subsets of E. A random finite set Ξ on E is defined as a measurable mapping

$$\Xi : \Omega \to \mathcal{F}(E). \tag{2.1}$$

This mapping induces a probability law for Ξ that can be equivalently described using a probability distribution function P_Ξ or a belief mass function β_Ξ. The probability distribution function $P_\Xi(\mathcal{T})$ is defined over the Borel subsets $\mathcal{T} \subseteq \mathcal{F}(E)$. The belief mass function $\beta_\Xi(S)$ is defined over the closed subsets $S \subseteq E$. When it exists, a probability density function $p_\Xi(W)$ for the random finite set Ξ is given by the Radon-Nikodym derivative

of its probability distribution function with respect to some dominating measure μ[1]

$$p_{\Xi}(W) = \frac{dP_{\Xi}}{d\mu}(W). \tag{2.2}$$

Alternatively, a density function $f_{\Xi}(W)$ can also be defined using the set derivative[2] of the belief mass function as

$$f_{\Xi}(W) = \frac{d\beta_{\Xi}}{dW}(\varnothing). \tag{2.3}$$

where $\varnothing$ denotes the empty set. In [36] it has been shown that these two densities are equivalent when units are ignored in the set derivative based density $f_{\Xi}(W)$. We also refer to this probability density function $f_{\Xi}(W)$ as the multitarget density function or the multitarget distribution function. The development of random finite set theory in the tracking literature has focused on using the belief mass function formulation and the corresponding finite set statistics (FISST). The FISST formulation allows an easy approach for analysis and manipulation of multitarget densities and other statistical descriptors of random finite sets without going into a detailed measure-theoretic description.

2.2.1 Set integral

Let Ξ be a random finite set on the space $\mathcal{X}$ and $f_{\Xi}(W)$ be its probability density function where $W \subseteq \mathcal{X}$. For a closed set S the integral (set integral) of $f_{\Xi}(W)$ is defined as

$$\int_S f_{\Xi}(W)\delta W \stackrel{\text{def}}{=} f_{\Xi}(\varnothing) + \sum_{n=1}^{\infty} \frac{1}{n!} \int_{S^n} f_{\Xi}(\{\mathbf{w}_1, \ldots, \mathbf{w}_n\}) d\mathbf{w}_1 \ldots d\mathbf{w}_n \tag{2.4}$$

$$= \beta_{\Xi}(S)\,, \tag{2.5}$$

where $S^n = S \times S \times \ldots n\,\text{times} \times S$. The notation δW denotes set integration. Since $f_{\Xi}(W)$ is a probability density function it must integrate to one

$$\int_{\mathcal{X}} f_{\Xi}(W)\delta W \stackrel{\text{def}}{=} f_{\Xi}(\varnothing) + \sum_{n=1}^{\infty} \frac{1}{n!} \int_{\mathcal{X}^n} f_{\Xi}(\{\mathbf{w}_1, \ldots, \mathbf{w}_n\}) d\mathbf{w}_1 \ldots d\mathbf{w}_n \tag{2.6}$$

$$= 1 \tag{2.7}$$

[1] For specific details about the measure μ for Borel subsets of $\mathcal{F}(E)$ please refer to [36].
[2] For definition and discussion on set derivatives please refer to [34, 35].

where $\mathcal{X}^n \stackrel{\text{def}}{=} \mathcal{X} \times \ldots n\,\text{times} \times \mathcal{X}$.

Since the realizations of the random finite set Ξ are sets with variable number of elements, a cardinality distribution function can be associated with it. The cardinality distribution function $\pi_\Xi(n)$ of the RFS Ξ which has the multitarget density function $f_\Xi(W)$ is given as

$$\pi_\Xi(n) = \text{Prob}(|\Xi| = n) \tag{2.8}$$

$$= \int_{|W|=n} f_\Xi(W)\delta W \tag{2.9}$$

$$= \frac{1}{n!}\int_{\mathcal{X}^n} f_\Xi(\{\mathbf{w}_1, \ldots, \mathbf{w}_n\})d\mathbf{w}_1 \ldots d\mathbf{w}_n. \tag{2.10}$$

Also let $M_\Xi(y)$ be the *probability generating function* (PGF) of the cardinality random variable $|\Xi|$ and is given by

$$M_\Xi(y) = \sum_{n=0}^{\infty} y^n\, \pi_\Xi(n). \tag{2.11}$$

2.2.2 Probability generating functional

Similar to the probability generating function defined for a random variable, we can associate an integral transform with the random finite set. Let $u(\mathbf{w})$ be a function mapping the state space $\mathcal{X}$ to $[0, 1]$. The *probability generating functional* (PGFL) of an RFS Ξ with multitarget density function $f_\Xi(W)$ is given as

$$G_\Xi[u] = \int_{\mathcal{X}} u^W\, f_\Xi(W)\delta W\,, \tag{2.12}$$

$$\text{where} \quad u^W \stackrel{\text{def}}{=} \prod_{\mathbf{w}\in W} u(\mathbf{w}) \quad \text{and} \quad u^\varnothing = 1. \tag{2.13}$$

The PGFL is a functional and is a mapping from functions to reals. The PGFL characterization of a random finite set is useful for calculating other statistical properties of the random finite set and for deriving filter update equations.

2.2.3 Probability hypothesis density

Random variables are commonly characterized using their mean, variance and other higher order moments. We would like to associate similar statistical quantities with the random

finite set. For random variables, moments are defined using either summation or integration. Since the addition operation is not naturally defined on sets, defining the expectation of an RFS in the traditional manner is not possible. An important and useful statistic of the RFS which can be defined using a modified definition of the first moment [34] is the *probability hypothesis density* (PHD) function. The PHD of an RFS Ξ with probability density function $f_\Xi(W)$ is:

$$D_\Xi(\mathbf{x}) = \int_{\mathcal{X}} f_\Xi(\{\mathbf{x}\} \cup W)\delta W. \tag{2.14}$$

Intuitively the value of the PHD function at $\mathbf{x} \in \mathcal{X}$ is equal to the cumulative density of all the sets which include the element $\mathbf{x}$. Since the PHD function is defined over the single element space $\mathcal{X}$, it provides a compact characterization of the random finite set. When the multitarget state is modelled as a random finite set, the peaks in its PHD function correspond to the unknown target states. The PHD has the property that its integration over the space $\mathcal{X}$ is equal to the expected cardinality

$$\int_{\mathcal{X}} D_\Xi(\mathbf{x})d\mathbf{x} = E(|\Xi|) = \sum_{n=0}^{\infty} n\,\pi_\Xi(n). \tag{2.15}$$

The second factorial moment can be similarly defined as

$$D_\Xi(\{\mathbf{x}_1, \mathbf{x}_2\}) = \int_{\mathcal{X}} f_\Xi(\{\mathbf{x}_1, \mathbf{x}_2\} \cup W)\delta W. \tag{2.16}$$

2.2.4 Change of variables formula

Let ζ be a function mapping finite dimensional sets to vectors in space $\mathcal{Y}$. For a real-valued function T and a multitarget distribution $f_\Xi(X)$ we have the following change of variables formula [33, Prop. 4, p. 180]

$$\int_{\mathcal{X}} T\left(\zeta(W)\right)\, f_\Xi(W)\delta W = \int_{\mathcal{Y}} T(\mathbf{y})\, Q_\Xi(\mathbf{y})d\mathbf{y} \tag{2.17}$$

where $Q_\Xi(\mathbf{y})$ is the probability distribution of the random vector $\zeta(\Xi)$ induced by the change of variables from the set W to the vector $\mathbf{y}$, $\mathbf{y} = \zeta(W)$. Note that the left hand side of (2.17) is a set integral whereas the right hand side is an ordinary integral. We frequently use the above change of variables formula to convert the intractable set integrals

into simpler ordinary integrals.

2.2.5 Campbell's theorem

Let $f_{\Xi}(X)$ be a multitarget density corresponding to the random finite set Ξ. Denote the PHD function and the second factorial moment density function of Ξ by $D_{\Xi}(\mathbf{x})$ and $D_{\Xi}(\{\mathbf{x}_1, \mathbf{x}_2\})$ respectively. Let $g(\mathbf{x})$ be a real vector-valued function defined over the space $\mathcal{X}$. Now consider the vector ξ defined as $\xi = \sum_{\mathbf{x}\in\Xi} g(\mathbf{x})$. Since Ξ is a random finite set, the vector ξ is a random vector. Then according to the quadratic version of Campbell's theorem [37, 38]

$$\mathbf{m}_{\xi} = E[(\xi)] = \int_{\mathcal{X}} g(\mathbf{x})\, D_{\Xi}(\mathbf{x}) d\mathbf{x} \tag{2.18}$$

$$\Sigma_{\xi} = E[(\xi - \mathbf{m}_{\xi})(\xi - \mathbf{m}_{\xi})^T] \tag{2.19}$$

$$= \int_{\mathcal{X}} g(\mathbf{x})\, g(\mathbf{x})^T D_{\Xi}(\mathbf{x}) d\mathbf{x} + \int_{\mathcal{X}} \int_{\mathcal{X}} g(\mathbf{x}_1)\, g(\mathbf{x}_2)^T \tilde{D}_{\Xi}(\{\mathbf{x}_1, \mathbf{x}_2\}) d\mathbf{x}_1 d\mathbf{x}_2\,, \tag{2.20}$$

$$\text{where} \quad \tilde{D}_{\Xi}(\{\mathbf{x}_1, \mathbf{x}_2\}) \stackrel{\text{def}}{=} D_{\Xi}(\{\mathbf{x}_1, \mathbf{x}_2\}) - D_{\Xi}(\mathbf{x}_1) D_{\Xi}(\mathbf{x}_2) \tag{2.21}$$

Thus the mean $\mathbf{m}_{\xi}$ and covariance matrix Σ_{ξ}, which represent the first and second order statistics of the random vector ξ, depend on the PHD function and the second factorial moment density function of the corresponding random set Ξ.

2.2.6 Union of mutually independent random finite sets

Let the random finite set Ξ be union of two statistically independent random finite sets Ξ_1 and Ξ_2 given as $\Xi = \Xi_1 \cup \Xi_2$. If $f_{\Xi_1}(W)$ and $f_{\Xi_1}(W)$ are multitarget densities of the random finite sets Ξ_1 and Ξ_2, then the multitarget density $f_{\Xi}(W)$ of Ξ is given by the following convolution relation [34],

$$f_{\Xi}(W) = \sum_{Y \subseteq W} f_{\Xi_1}(Y)\, f_{\Xi_2}(W - Y). \tag{2.22}$$

The PGFLs and the PHDs of the corresponding random finite sets are related as

$$G_{\Xi}[u] = G_{\Xi_1}[u]\, G_{\Xi_2}[u] \tag{2.23}$$

$$D_{\Xi}(\mathbf{x}) = D_{\Xi_1}(\mathbf{x}) + D_{\Xi_2}(\mathbf{x}). \tag{2.24}$$

The above result can be easily proved using the properties of probability generating functional and the basic rules for functional derivatives (Chap. 11, [34]). We also have the result

$$\tilde{D}_{\Xi}(\{\mathbf{x}_1, \mathbf{x}_2\}) = \tilde{D}_{\Xi_1}(\{\mathbf{x}_1, \mathbf{x}_2\}) + \tilde{D}_{\Xi_2}(\{\mathbf{x}_1, \mathbf{x}_2\}) \tag{2.25}$$

$$\text{where} \quad \tilde{D}_{\Xi}(\{\mathbf{x}_1, \mathbf{x}_2\}) \stackrel{\text{def}}{=} D_{\Xi}(\{\mathbf{x}_1, \mathbf{x}_2\}) - D_{\Xi}(\mathbf{x}_1) D_{\Xi}(\mathbf{x}_2). \tag{2.26}$$

2.3 Examples of random finite sets

2.3.1 Poisson RFS

A Poisson random finite set can be completely characterized by its probability hypothesis density function $D_{\Xi}(\mathbf{x})$. Let the mean cardinality of the Poisson RFS be μ, then

$$\mu = \int_{\mathcal{X}} D_{\Xi}(\mathbf{x}) d\mathbf{x}. \tag{2.27}$$

The multitarget distribution of the Poisson random finite set is given by

$$f_{\Xi}(W) = e^{-\mu} \prod_{\mathbf{x} \in W} D_{\Xi}(\mathbf{x}). \tag{2.28}$$

The cardinality distribution of the Poisson random finite set is Poisson with mean μ. Samples of a Poisson RFS can be generated by first sampling the cardinality of the set and then sampling elements of the set from the normalized PHD function $s_{\Xi}(\mathbf{x}) = \frac{D_{\Xi}(\mathbf{x})}{\mu}$. The probability generating functional of a Poisson RFS is

$$G_{\Xi}[u] = e^{\langle D_{\Xi}, u \rangle - \mu}, \tag{2.29}$$

where for functions $a(\mathbf{x})$ and $b(\mathbf{x})$ the notation $\langle a, b \rangle$ is defined as

$$\langle a, b \rangle \stackrel{\text{def}}{=} \int_{\mathcal{X}} a(\mathbf{x})\, b(\mathbf{x}) d\mathbf{x}. \tag{2.30}$$

The second factorial moment of a Poisson RFS is

$$D_{\Xi}(\{\mathbf{x}_1, \mathbf{x}_2\}) = D_{\Xi}(\mathbf{x}_1)\, D_{\Xi}(\mathbf{x}_2). \tag{2.31}$$

2.3.2 Independent and identically distributed cluster RFS

An *independent and identically distributed cluster* (IIDC) RFS can be completely specified using its cardinality distribution $\pi_\Xi(n)$ and its PHD function $D_\Xi(\mathbf{x})$. Let the mean cardinality of the IIDC RFS be μ and the normalized PHD function be $s_\Xi(\mathbf{x}) = \frac{D_\Xi(\mathbf{x})}{\mu}$. The probability density function of an IIDC RFS is given by

$$f_\Xi(W) = |W|!\,\pi_\Xi(|W|) \prod_{\mathbf{x}\in W} s_\Xi(\mathbf{x}). \tag{2.32}$$

The IIDC RFS is a generalization of the Poisson RFS where the cardinality distribution is allowed to be arbitrary. If $M_\Xi(y)$ is the PGF of the cardinality distribution $\pi_\Xi(n)$, then the PGFL of an IIDC RFS is given as

$$G_\Xi[u] = M_\Xi(\langle s_\Xi, u\rangle). \tag{2.33}$$

The second factorial moment of an IIDC RFS can be calculated to be

$$D_\Xi(\{\mathbf{x}_1, \mathbf{x}_2\}) = a\, s_\Xi(\mathbf{x}_1)\, s_\Xi(\mathbf{x}_2)\,, \tag{2.34}$$

where

$$a = \sum_{n=0}^{\infty} n(n-1)\,\pi_\Xi(n). \tag{2.35}$$

2.3.3 Bernoulli RFS

A Bernoulli random finite set Ξ can either be an empty set with probability $1-r$ or a singleton set $\{\mathbf{x}\}$ with probability r. The singleton element $\mathbf{x}$ when present is drawn from the density function $s_\Xi(\mathbf{x})$. The probability density function of a Bernoulli RFS is

$$f_\Xi(W) = \begin{cases} 1-r, & \text{if } W = \varnothing \\ r\, s_\Xi(\mathbf{x}), & \text{if } W = \{\mathbf{x}\} \\ 0 & \text{otherwise.} \end{cases} \tag{2.36}$$

The probability generating functional of a Bernoulli RFS is given by

$$G_{\Xi}[u] = 1 - r + r\,\langle s_{\Xi}, u\rangle. \tag{2.37}$$

The probability hypothesis density of a Bernoulli RFS is given by

$$D_{\Xi}(\mathbf{x}) = r\,s_{\Xi}(\mathbf{x}). \tag{2.38}$$

Since a Bernoulli RFS can only be an empty set or a singleton set, its second and higher order factorial moments are zero.

2.3.4 Multi-Bernoulli RFS

A multi-Bernoulli RFS Ξ is defined as the union of N independent Bernoulli random finite sets, as $\Xi = \Xi_1 \cup \Xi_2 \cup \cdots \cup \Xi_N$, where the random finite sets Ξ_i are Bernoulli random finite sets with parameters given by $\{r_i, s_i(\mathbf{x})\}$ for $i = 1, 2, \ldots, N$. The probability density function of a multi-Bernoulli RFS [39] is,

$$f_{\Xi}(W) = \begin{cases} \prod_{j=1}^{N}(1 - r_j), & \text{if } W = \varnothing \\ \prod_{j=1}^{N}(1 - r_j) \times \sum_{1 \le i_1 \ne \cdots \ne i_n \le N} \prod_{j=1}^{n} \frac{r_{i_j}\, s_{i_j}(\mathbf{x}_j)}{1 - r_{i_j}}, & \text{if } W = \{\mathbf{x}_1, \mathbf{x}_2 \ldots \mathbf{x}_n\},\ n \le N \\ 0 & \text{otherwise.} \end{cases} \tag{2.39}$$

The PGFL of the multi-Bernoulli RFS is given by

$$G_{\Xi}[u] = \prod_{i=1}^{N}(1 - r_i + r_i\,\langle s_i, u\rangle). \tag{2.40}$$

The PHD of the multi-Bernoulli RFS (Ex. 91, Chap. 16, [34]) is

$$D_{\Xi}(\mathbf{x}) = \sum_{i=1}^{N} r_i\, s_i(\mathbf{x}). \tag{2.41}$$

The second factorial moment of the multi-Bernoulli RFS can be calculated to be

$$D_{\Xi}(\{\mathbf{x}_1, \mathbf{x}_2\}) = \sum_{i=1}^{N} \sum_{j=1, j\neq i}^{N} r_i \, r_j \, s_i(\mathbf{x}_1) \, s_j(\mathbf{x}_2) \tag{2.42}$$

$$= D_{\Xi}(\mathbf{x}_1) \, D_{\Xi}(\mathbf{x}_2) - \sum_{i=1}^{N} r_i^2 \, s_i(\mathbf{x}_1) \, s_i(\mathbf{x}_2). \tag{2.43}$$

2.4 Sensor measurement models

In this thesis we consider two types of sensor measurement models. Following the nomenclature used by Mahler in [14], we refer to them as the *standard sensor model* and the *superpositional sensor model*. We now discuss these two models and provide some examples of them.

2.4.1 Standard sensor model

A large number of sensors that have been studied in the literature fall under the category of "standard" sensors. Measurements gathered by these sensors are either generated by individual targets or by the clutter process.

Let Υ_k be the set of target generated measurements and let Λ_k be the set of clutter measurements at time k. The complete set of measurements gathered by a standard sensor at time k can be written as $Z_k = \Upsilon_k \cup \Lambda_k$. Given the set of measurements Z_k, the origin of individual measurements within the set Z_k is not known. Associated with each sensor is its probability of detection function. If a target with state $\mathbf{x}$ is present, then the sensor detects the target with probability of detection $p_d(\mathbf{x})$.

Let $Z_k = \{\mathbf{z}_1, \mathbf{z}_2, \dots \mathbf{z}_m\}$ be the set of measurements generated by a sensor at some time instant k. The assumptions made by the standard sensor model are as follows:

1. Each target can generate at most one measurement.

2. Each measurement $\mathbf{z}_i, i \in [\![1, m]\!]$, is either associated with a target (i.e. $\mathbf{z}_i \in \Upsilon_k$) or is a clutter measurement (i.e. $\mathbf{z}_i \in \Lambda_k$).

3. A single target with state $\mathbf{x}$ is detected by the sensor with probability of detection $p_d(\mathbf{x})$ or missed with probability $1 - p_d(\mathbf{x})$.

4. The measurements corresponding to targets are independent conditional on the multitarget state.

5. The set of clutter measurements (Λ_k) and the set of target generated measurements (Υ_k) are statistically independent.

When a target with state $\mathbf{x}$ is detected, the likelihood that it generates a measurement $\mathbf{z}$ at time k is denoted by $h_k(\mathbf{z}|\mathbf{x})$. When multiple standard sensors are present, each of them generate measurements following the assumptions listed above. Additionally, conditional on the multitarget state, measurements generated by different sensors are assumed independent.

Examples of sensor systems that can be modeled using this measurement model are

1. Radar systems [16],

2. Sonar systems [17],

3. Range-and-bearing sensors [18].

As a specific example consider a bearings-only measurement sensor that gathers angle information about targets in the $x-y$ plane. It records angle values in degrees in the range $[0, 360)$ with respect to a reference axis. At some time step k assume that there are three targets of which the bearings sensor detects two of them and records the target generated measurement set as $\Upsilon_k = \{11.2, 56.5\}$. In addition let there be four clutter measurements recorded by the sensor and are given by $\Lambda_k = \{102.1, 33, 270.8, 140.1\}$. The combined measurement set generated by the sensor is then given as $Z_k = \{102.1, 11.2, 33, 56.5, 270.8, 140.1\}$. Note that the sensor records measurements in no particular order and there is no information available about the generation mechanism (target-generated or clutter) of individual measurements. The filtering algorithms we develop should consider this inherent uncertainty in sensor measurements.

2.4.2 Superpositional sensor model

The standard sensors form an important class of sensors but are not exhaustive. Another class of sensors that we study in this thesis are the "superpositional" sensors. In the standard sensor model, the measurement generated by each sensor is modeled as a set. In

the superpositional sensor model we assume that each sensor gathers a measurement vector of fixed and known dimensions. Let $\mathbf{Z}_k$ be the combined measurement vector obtained by appending the measurement vectors of all the sensors together. The superpositional sensor model makes the following assumptions:

1. Each measurement is affected by multiple targets in an additive fashion, i.e., the contribution to a measurement due to multiple targets is equal to the sum of contributions to that measurement from each of the targets when present alone.

2. Measurements gathered by different sensors are correlated because each target can potentially affect any number of sensors.

Many sensors belong to the category of superpositional sensors. Examples include

1. Direction-of-arrival sensors for linear antenna arrays [11],

2. Antenna arrays in multi-user detection for wireless communication networks [12],

3. Multipath channel modeling in MIMO-OFDM channels [13],

4. Acoustic amplitude sensors [10],

5. Radio-frequency tomographic tracking systems [5].

If X_k is the multitarget state at time k, then the multisensor multitarget likelihood function is denoted by $h_k(\mathbf{Z}_k|X_k)$. In the case of superpositional sensors the likelihood function $h_k(\mathbf{Z}_k|X_k)$ has the following form,

$$h_k(\mathbf{Z}_k|X_k) = h_k\left(\mathbf{Z}_k|\zeta(X_k)\right) \tag{2.44}$$

$$= h_k\left(\mathbf{Z}_k|\sum_{\mathbf{x}\in X_k} g(\mathbf{x})\right) \tag{2.45}$$

where ζ and g are (potentially non-linear) functions mapping to vectors of reals. The function ζ operates on the random finite set whereas the function g operates on the target states that are members of the set.

In this thesis we focus on the case where the likelihood function has the following Gaussian form:

$$h_k\left(\mathbf{Z}_k|\sum_{\mathbf{x}\in X_k} g(\mathbf{x})\right) = \mathcal{N}_{\Sigma_\mathbf{z}}\left(\mathbf{Z}_k - \sum_{\mathbf{x}\in X_k} g(\mathbf{x})\right) \tag{2.46}$$

where the notation $\mathcal{N}_\Sigma(\mathbf{x})$ denotes evaluation at $\mathbf{x}$ of a zero-mean Gaussian distribution with covariance matrix Σ. We will use the notation $\Sigma_\mathbf{z}$ throughout to denote the covariance of the measurement noise. Although the Gaussian noise assumption is not essential for deriving update equations, it is an important contributing factor for computationally tractable approximations.

Example: Acoustic amplitude sensors

Acoustic sensor networks can be used for multitarget tracking based on the strength of the emitted acoustic signals. In this thesis we adapt the acoustic amplitude sensor measurement model discussed in [10]. It is an active tracking system in which each target emits an acoustic signal of known amplitude A and all the sensors receive the signal. If a target at location $\mathbf{x}$ emits the acoustic signal, a sensor located at d^j receives the signal at a reduced strength of $g^j(\mathbf{x}) = A/\max(\|\mathbf{x} - d^j\|, d_0)^\kappa$ where $\|\mathbf{x}\|$ denotes the Euclidean norm of vector $\mathbf{x}$; κ is the path loss exponent and d_0 is the threshold distance such that the received signal amplitude saturates if the target is closer than distance d_0 of the sensor. When multiple targets are present, the strength of the combined signal received by each of the sensors is the sum of the strength of the signals due to each of the individual targets. Thus the measurement received by sensor j at time k, z_k^j, can be modelled as:

$$z_k^j = \zeta^j(X_k) + v_k^j \tag{2.47}$$

$$= \sum_{\mathbf{x}\in X_k} \frac{A}{\max(\|\mathbf{x} - d^j\|, d_0)^\kappa} + v_k^j \tag{2.48}$$

where v_k^j is the zero-mean Gaussian measurement noise. A limitation of the above model is that though the signal amplitude received by a sensor from a target saturates as the target approaches the sensor, the combined signal strength received by a sensor can still be large if multiple targets are in close vicinity. A more realistic model would be to saturate the combined signal strength received by a sensor. For simplicity we use the model given by

(2.48) in this thesis.

2.5 The Bayes filter

We solve the problem of sequential multitarget state estimation using the recursive Bayes filter. The Bayes filter provides a framework for data fusion and state estimation but its direct implementation is often analytically and computationally intractable. The various contributions of this thesis are to develop approximations to the Bayes filter which are analytically and computationally tractable. Although here we discuss the Bayes filter for random sets [34], the filter equations for the case of random vectors [40] are similar where instead of the set integrals we have vector integrals.

At time k, given all the measurements up to time k, the complete system state is specified by the multitarget posterior distribution $f_{k|k}(X_k|Z_{1:k})$. The Bayes filter solves for the posterior distribution in a recursive manner. Let $\Theta_{k+1|k}(X_k|W)$ be the multitarget transition function at time $k+1$. The prediction step of the filter incorporates the target motion model and is given by the Chapman-Kolmogorov equation

$$f_{k+1|k}(X_{k+1}|Z_{1:k}) = \int_{\mathcal{X}} \Theta_{k+1|k}(X_k|W) f_{k|k}(W|Z_{1:k})\, \delta W\,. \tag{2.49}$$

Let $L_{k+1}(Z_{k+1}|X_{k+1})$ be the multitarget multisensor likelihood function at time $k+1$. The update step of the filter uses the measurements at time $k+1$, the multitarget likelihood function and applies Bayes' rule to obtain the posterior at time $k+1$ as

$$f_{k+1|k+1}(X_{k+1}|Z_{1:k+1}) = \frac{L_{k+1}(Z_{k+1}|X_{k+1}) f_{k+1|k}(X_{k+1}|Z_{1:k})}{\int_{\mathcal{X}} L_{k+1}(Z_{k+1}|W) f_{k+1|k}(W|Z_{1:k})\, \delta W}. \tag{2.50}$$

In a traditional filter implementation the sets of the multitarget state X_{k+1} and measurements $Z_{1:k}$ are modelled as vectors $\mathbf{X}_{k+1}$ and $\mathbf{Z}_{1:k}$, respectively, and the set integrals in (2.49) and (2.50) are replaced with vector integrals. For both vector and set based modelling the above equations involve evaluation of integrals which cannot be analytically solved except for a few special cases.

2.6 Gaussian distributions

We make a note of some results related to Gaussian distributions [41] which will be frequently used in later parts of the thesis. Let $\mathbf{x}$ and $\mathbf{z}$ be vectors of possibly different dimensions. Let $\mathbf{m_x}$ be a mean vector and $\Sigma_\mathbf{x}$ and $\Sigma_\mathbf{z}$ be the covariance matrices of appropriate dimensions. Let H be a matrix such that $\mathbf{z}$ and $H\mathbf{x}$ have the same dimensions. Then we have

$$\mathcal{N}_{\Sigma_\mathbf{z}}(\mathbf{z} - H\mathbf{x})\,\mathcal{N}_{\Sigma_\mathbf{x}}(\mathbf{x} - \mathbf{m_x}) = \mathcal{N}_{\Sigma_\mathbf{z} + H\Sigma_\mathbf{x}H^T}(\mathbf{z} - H\mathbf{m_x})\,\mathcal{N}_{\tilde{\Sigma}_\mathbf{x}}(\mathbf{x} - \tilde{\mathbf{m}}_\mathbf{x}), \tag{2.51}$$

$$\text{where} \quad \tilde{\mathbf{m}}_\mathbf{x} = \mathbf{m_x} + K(\mathbf{z} - H\mathbf{m_x}) \tag{2.52}$$

$$\tilde{\Sigma}_\mathbf{x} = (I - KH)\Sigma_\mathbf{x} \tag{2.53}$$

$$K = \Sigma_\mathbf{x}H^T(\Sigma_\mathbf{z} + H\Sigma_\mathbf{x}H^T)^{-1}. \tag{2.54}$$

In these equations H^T denotes the transpose of matrix H and I is the identity matrix of appropriate dimension. Let $\mathbf{m}_1$ and $\mathbf{m}_2$ be the means and Σ_1 and Σ_2 be the covariance matrices of two Gaussian densities. Then we have

$$\int \mathcal{N}_{\Sigma_1}(\mathbf{x} - \mathbf{m}_1)\,\mathcal{N}_{\Sigma_2}(\mathbf{x} - \mathbf{m}_2)d\mathbf{x} = \mathcal{N}_{\Sigma_1+\Sigma_2}(\mathbf{m}_1 - \mathbf{m}_2). \tag{2.55}$$

2.7 The error metric

In order to compare the performance of different multitarget tracking algorithms, we need an error metric to quantify the difference between the sets of true multitarget state and the estimated multitarget state. Since sets are involved, a root mean squared type of metric cannot be applied. We use the *optimal subpattern assignment* (OSPA) error metric developed by Schuhmacher et al. in [42]. This metric is specifically designed for performance evaluation of multi-object filters. The OSPA metric penalizes the cardinality error in the estimates using the cardinality penalty factor c. Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$ and $Y = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$ be any arbitrary sets then, for $n \geq m$, the OSPA metric between the sets X and Y is defined

as

$$d_p^{(c)}(X,Y) \stackrel{\text{def}}{=} \left(\frac{1}{n} \min_{\pi \in \Pi} \sum_{i=1}^{m} d^{(c)}(\mathbf{x}_i, \mathbf{y}_{\pi(i)})^p + c^p(n-m) \right)^{1/p}, \tag{2.56}$$

$$\text{where} \quad d^{(c)}(\mathbf{x},\mathbf{y}) \stackrel{\text{def}}{=} \min\{d(\mathbf{x},\mathbf{y}), c\} \tag{2.57}$$

where Π is the set of possible permutations of the set $\{1, 2, \ldots, n\}$, $d(\mathbf{x},\mathbf{y})$ is any distance metric between the vectors $\mathbf{x}$ and $\mathbf{y}$ and p is a fixed parameter. When $m > n$, we calculate the metric $d_p^{(c)}(Y,X)$. The OSPA metric finds the best permutation of the larger set which minimizes its distance from the smaller set and assigns a fixed penalty for each cardinality error. In our evaluation of the OSPA metric we choose $d(\mathbf{x},\mathbf{y})$ to be the Euclidean distance metric.

2.8 Sensor networks for tracking

Simultaneous detection, localization and tracking of multiple humans has numerous applications in areas such as security, surveillance, and healthcare. Many different tracking systems [43] have been developed to solve this problem which make use of different kinds of sensor hardware. Some of the sensor measurement modalities that have been discussed in the literature are optical images and video [44, 45], infrared sensors [46, 47], acoustic sensors [48, 49], and radio sensors [5, 50, 51]. Many of the sensing methods require the targets to carry a device which assists in the tracking procedure [50, 52] which may not be always possible. In this thesis we are interested in device-free target tracking using radio-frequency sensor networks.

Radio-frequency tomography is the process of monitoring an area to detect mobile targets based on the additional attenuation and fluctuations they cause in wireless transmissions. It is a device-free method and the targets are not required to carry any additional device for tracking purpose. A good overview of the challenges faced in realizing a device-free system and the different techniques for target localization using RF sensor networks can be found in the review paper [53]. In the next section we review various solutions that have been proposed for target tracking using RF sensor networks.

2.8.1 Radio-frequency based tracking

Zhang et al. developed an indoor multitarget tracking system for ceiling-mounted RF sensor nodes based on the interference caused by moving objects [54–56]. These systems achieve good tracking performance but require calibration and are restricted to the ceiling deployment of sensors. This is not practical in several important scenarios such as search-and-rescue operations and military surveillance.

Youssef et al. [57] proposed to use existing WiFi devices for device-free tracking in indoor environments. They demonstrated single target detection and tracking [57, 58] based on signal strength maps constructed during an offline training phase. Due to limited number of devices the tracking accuracy is low and frequent system calibration may be required to account for the dynamic multi-path effects in indoor environments.

Most previous RF tomography tracking techniques [5–8] have focused on single target tracking. In [5, 6], Wilson and Patwari proposed an inverse imaging algorithm which first obtains an attenuation map and then applies a Kalman filter to track the peak in the map. A compressed sensing based imaging algorithm is proposed in [59] for inference in RF sensor networks. Imaging algorithms have the drawback that the problem of inverse imaging is in general computationally challenging and the monitoring region is divided into a grid of pixels which limits its accuracy. Through-wall tracking is demonstrated in [6] using a network of 34 RF sensor nodes. Li et al. introduced a new measurement model based on experimental data and used it in a sequential Monte Carlo algorithm for tracking [7]. This method incorporated online Expectation-Maximization so that model parameters could be learned during the tracking task. This approach was extended in [8] to simultaneously estimate the locations of the sensor nodes.

A multitarget extension of the RF tomography measurement model in [7,8] was proposed by Thouin et al. [4]. The measurements are modelled to have a superpositional form (see Section 2.4.2) and performance assessment was based entirely on simulation; no practical experimentation was conducted.

Most of the experimental work in [5, 7, 8] has been restricted to outdoor environments. A variance based algorithm was proposed in [6] for through-wall tracking but is limited to moving targets and cannot accurately localize stationary targets. More recently in [60], Wilson and Patwari developed a skew-Laplace signal strength model for indoor target tracking. Received signal-strength measurements are modeled using skew-Laplace distributions

whose parameters are experimentally obtained through training measurements. The parameters change depending on whether the target is close to the line-of-sight between the sensors. Successful tracking of two targets was reported in [60], but the case of an unknown and time-varying number of targets was not addressed. The model parameters need to be trained using a target with known position; Wilson and Patwari suggest that parameters learned in one environment can be applied successfully in another.

2.8.2 Tracking algorithms

In Chapter 3, for performing RF tomography using real sensor network data, we assume that the number of targets is known and the measurements are vectors of known dimension. In this setup many of the existing multitarget tracking filter can be directly utilised. Because of the highly non-linear measurement model, classical filters such as the Kalman filter [61], the extended Kalman filter [40], and the unscented Kalman filter [62, 63] do not perform well. Particle based filters such as sequential Monte Carlo methods [1, 64–68] and Markov chain Monte Carlo methods [3, 69–71] can be used to propagate a particle approximation of the posterior distribution. Particle filters have been used for RF tomography tracking in [7, 8, 60].

As the number of targets increase the number of particles required to efficiently sample the multitarget state space grows exponentially [72]. One of the strategies to avoid this exponential growth is to use the *multiple particle filter* (MPF) [2] proposed by Bugallo et al. which uses one low-dimensional particle filter for each target. Since each of the particle filters operates in a much smaller dimension, fewer particles are required and the increase in the number of particles is linear with increase in the number of targets. The likelihood function in general can be a function of all the target states hence the different particle filters need to interact to perform the weight update. The interactions allow the filters to exchange information such as statistical moments or particles. Several extensions and improvements of the MPF filter [73–77] have been developed which propose different interaction mechanisms to improve the accuracy of the individual particle filters.

2.9 Multitarget tracking using standard sensors

The standard sensor model is discussed in Section 2.4.1. Since the observations in many practical sensor systems [16, 18, 40, 78] such as radar, sonar, and range-and-bearings sensor

systems follow this measurement model, it has been well researched and numerous filters have been proposed. We discuss the filters proposed in the literature within the traditional filtering framework and the random finite set framework.

2.9.1 Traditional filters

The traditional approach to multitarget tracking using standard sensors has been to first perform a measurement-to-target association and then apply the Kalman filter [61]. Some of the successful filters are the joint probabilistic data association (JPDA) filter [18, 31] and the multiple hypothesis tracking (MHT) filter [32, 78] and many of their variants. The JPDA filter assumes fixed and known number of targets and calculates probabilities for different possible associations between the measurements and targets. The average of measurements with association probabilities as weights is used in the update stage to obtain the posterior. The MHT filter [32, 78] propagates a set of competing measurement-to-target association hypothesis over time. The MHT filter can track a time varying number of targets. Particle filter based solutions to the problem of data association and multitarget tracking have also been developed in the literature [79–82]. Since the number of associations grow exponentially with the number of targets, measurements and sensors, traditional filters usually have high computational requirements. Heuristics such as gating are often used to limit the computations.

2.9.2 Random finite set based filters: Single sensor

We now provide a background on random finite set based multitarget tracking filters which do not require explicit measurement-to-target association. The random finite set framework for multitarget tracking was introduced by Goodman et al. in [33]. A detailed treatment of random finite set and the finite set statistics can be found in [33–35].

The majority of research based on random finite set theory has focused on single sensor multitarget tracking. The optimal filter is the recursive Bayes filter discussed in Section 2.5. But direct implementation of this filter suffers from analytical intractability and high computational requirements. Some of the direct implementations of the recursive Bayes filter are based on particle filter approximations [36, 83–85] but they are only practical when very few targets are present.

Moment filters

Several approximate filters have been proposed in the literature that propagate some sufficient statistics, such as the PHD and cardinality distribution, of the random finite set instead of propagating the full multitarget density. The *probability hypothesis density* (PHD) filter [86] propagates, over time, the probability hypothesis density function which is defined over the single target state space. Improving on the PHD filter, the *cardinalized probability hypothesis density* (CPHD) filter [87] propagates the distribution of the number of targets (the cardinality) in addition to the probability hypothesis density function.

Multi-Bernoulli filters

Both the PHD and CPHD filter use a single PHD function to represent the multiple single target states. In contrast, the multi-Bernoulli filter [34, 39] models each target state with a Bernoulli random finite set. Thus each target has a scalar existence probability and a state density function. This allows more accurate state representation and also provides easy track maintenance [34]. Generalizations of the multi-Bernoulli filter have also been proposed such as the labeled multi-Bernoulli filter [88] and the generalized labeled multi-Bernoulli filter [89] which can track target labels over time.

Implementations

Various implementations of the PHD, CPHD and multi-Bernoulli filters have been proposed in the literature. The Gaussian mixture-based implementations [39, 41, 90–94] provide closed form filter equations under linear-Gaussian assumptions. Extensions to non-linear models using extended Kalman filter and unscented Kalman filter approaches have also been proposed [39, 41, 90]. For general non-linear models several sequential Monte Carlo based implementations [36, 39, 95–98] have been proposed. Convergence results for the various filter implementations have also been discussed in the literature [99–103].

Many improvements have been proposed for the sequential Monte Carlo implementations [104–107] to improve the filter performance. In [104, 106] explicit association is maintained between particles and measurements to provide improvements in cardinality and state estimates. Measurement-driven target birth intensity is used in [107] to improve the estimation accuracy of PHD and CPHD filters. An auxiliary particle filter implementation of the PHD filter is proposed in [105] where proposal distributions are designed to

efficiently migrate particles towards high likelihood regions.

These algorithms and their extensions have been successfully applied to the problem of single sensor multitarget tracking in the presence of clutter using both simulated measurements and real measurements. Specifically they have been used in direction-of-arrival tracking [11], sonar image tracking [17], audio signal processing [108], group target tracking [109], ground moving target tracking [110], extended target tracking [111, 112], navigation and map building [113, 114], and visual image tracking [115–117].

All the above filters use a single kind of random finite set to model the multitarget state. We can gain by modeling the multitarget state as a union of different kinds of random finite sets. A hybrid of the multi-Bernoulli filter and the PHD filter is developed by Williams [118, 119]. In [118, 119] a Poisson RFS is used to model new targets and targets with low probability of existence. This results in fast track initiation and use of fewer Bernoulli components. Pollard et al. in [120] used a hybrid combination of the MHT filter and the Gaussian mixture CPHD (GM-CPHD) filter for multitarget tracking. The GM-CPHD filter provides a robust cardinality estimate of the multitarget state which is complemented by accurate state estimates from the MHT filter. A combination of the MHT and PHD filter is utilized by Panta et al. to obtain track-valued estimates [121]. In [121] the PHD filter is used as a clutter filter by utilising its output to gate the input for the MHT filter. Although hybrid filters have been developed in the literature for multitarget tracking, they are restricted to the case of standard sensors and cannot process superpositional observations.

2.9.3 Random finite set based filters: Multiple sensors

A review of the different multisensor multitarget tracking algorithms for the standard sensor model based on random finite set theory can be found in [35, Ch. 10]. A multisensor extension of the PHD filter was first derived for the case of two sensors by Mahler [19, 20]. The filter equations were further generalized to include an arbitrary number of sensors by Delande et al. [122]. Braca et al. [123] proved that asymptotically as the number of sensors goes to infinity the PHD behaves like a mixture of Gaussians, with Gaussian components centered at maximum likelihood target state estimates.

The exact filter update equations of the general multisensor PHD filter are not computationally tractable except for a few simple cases. In [124, 125] Delande et al. simplify

the filter update equations when the fields of view of different sensors have limited overlap. This reduces the computational complexity to some extent and a particle filter based implementation is discussed by the authors. Jian et al. in [126] suggest to implement the general multisensor PHD filter by repeated application of the two sensor PHD filter [19]. Implementation details for realizing the general multisensor PHD filter in this manner are not provided and the numerical simulations are restricted to the case of two sensors.

To avoid the combinatorial computational complexity of the general multisensor PHD filter, some approximate multisensor filters have been proposed in the literature. The *iterated-corrector PHD filter* [20] processes multisensor information in a sequential manner. A single sensor PHD filter processes measurements from the first sensor. Using the output PHD function produced by this step as the predicted PHD function, another single sensor PHD filter processes measurements from the second sensor and so on. As a result the final output depends on the order in which sensors are processed [127]. This dependence on the sensor order can be mitigated by employing the *product multisensor PHD and CPHD filters* proposed by Mahler [21]. Although the final results are independent of sensor order, Ouyang and Ji [128] have reported that Monte Carlo implementations of the product filters are unstable and the problem worsens as the number of sensors increases.

In Chapter 6 we develop a trellis based greedy algorithm to implement the general multisensor CPHD filter. Hence here we discuss other trellis based algorithms that have been developed for target tracking. For single-sensor single-target tracking, the Viterbi algorithm is applied over a trellis of measurements constructed over time in [129]. Each column of the trellis is a measurement scan at a different time step. The Viterbi algorithm is used to find the best path in the trellis corresponding to data associations over time. This approach has been extended for multitarget tracking in [130] for a fixed and known number of targets. The nodes of the trellis correspond to different data association hypotheses and the transition weights are based on measurement likelihoods. The Viterbi algorithm was also applied in [131], in conjunction with energy based transition weights, to identify the K-best non-intersecting paths over the measurement trellis when K targets are present.

The update equations in the general multisensor PHD/CPHD filters are similar to the update equations of the single sensor PHD/CPHD filters for extended targets [111, 132]. The similarity is in the sense that for extended targets the update equation requires partitioning of the single sensor measurement set which can be computationally demanding. Granstrom et al. [112] proposed a Gaussian mixture model-based implementation of the

PHD filter for extended targets with reduced partitioning complexity. This is done by calculating the Mahalanobis distance between the measurements and grouping together measurements which are close to each other within a certain threshold. Orguner et al. [132] used a similar method to reduce computations in the Gaussian mixture model-based implementation of the CPHD filter for extended targets.

2.10 Multitarget tracking using superpositional sensors

The superpositional sensor model is discussed in Section 2.4.2. Examples of sensor systems which generate superpositional observations are antenna arrays in wireless communication, acoustic amplitude sensor networks, and radio-frequency sensor networks. The problem of multitarget tracking using superpositional sensors is similar to the problem of signal parameter estimation [133] from sensor array observations and is related to the class of track-before-detect problems [78].

2.10.1 Traditional filters

In the signal parameter estimation problem, signals originating from multiple sources are combined in a linear fashion and corrupted by noise. The objective is to estimate the number of sources and their parameters [133]. Many algorithms [133–136] have been developed in the literature to solve this problem. These methods require some quasi-stationarity assumption to hold in order to estimate the sample covariance matrix of the observations. In the multitarget tracking setup which we investigate, the target states and hence the observations are constantly evolving over time and hence these algorithms cannot be directly used.

Some algorithms perform a preprocessing step to convert superpositional or correlated sensor measurements into appropriate standard sensor measurements [78]. These set-valued measurements are then filtered using the numerous existing algorithms available to process standard sensor measurements such as JPDA, MHT and particle filters. The track-before-detect problem is to perform tracking directly on the raw measurements without any preprocessing step. The preprocessing step typically involves extracting important signal features and thresholding them to obtain set-valued measurements. Since the raw observations are preprocessed, this approach can lead to information loss and result in poor tracking performance especially when the signal-to-noise ratio is low.

The multiple particle filters have been applied to the problem of superpositional sensors in [73–76] but their applicability is limited because they require the number of targets to be fixed and known.

2.10.2 Random finite set based filters

Particle implementations of the recursive Bayes filter based on RFS modelling have been developed for the case of superpositional sensors [12,13,137–139]. A bootstrap particle filter is used in [12, 137] for multi-user detection but simulations are limited to a maximum of three active users. For higher number of users, the efficiency of the bootstrap particle filter significantly reduces since sampling the space of finite sets is computationally expensive. A Rao-Blackwellized particle filter implementation is used in [13,138] for estimating multi-path channel state in wireless MIMO-OFDM systems. Although the Rao-Blackwellized particle filter implementation is computationally more efficient than the bootstrap particle filter implementation, it relies on the assumption that the system is conditionally linear and Gaussian.

The random finite set based filters have also been extended to other non-standard models such as extended target tracking [111,112,132], unresolved target tracking [140,141], and tracking from pixelized images [116, 142].

Moment filters

In [11,108,143] the superpositional measurements are converted to standard measurements which are then processed by the PHD and CPHD filters for standard sensors. The measurement model in [11], for direction-of-arrival tracking using sensor arrays, has a superpositional form but the measurements are converted into separable set-valued measurements using Fourier spectrum analysis of observations. The PHD filter for standard sensors is then applied for tracking. Similarly in [143] the acoustic sensor array observations have a superpositional form but the acoustic measurements are first converted into direction-of-arrival measurements using the multiple signal classification algorithm [133]. A grid based particle implementation of the PHD and CPHD filters for standard sensors is used to perform multitarget tracking.

The exact CPHD filter equations for superpositional sensors were first derived by Mahler [14] but are computationally intractable. An analytically tractable closed form

Gaussian mixture model-based implementation of the CPHD filter for superpositional sensors was derived by Hauschildt [15]. Though analytically tractable, its direct implementation is computationally demanding. An approximate but computationally tractable PHD filter for superpositional sensors was developed by Thouin et al. [4]. The derivation of this approximate filter relied on Campbell's theorem [37] and Gaussian approximation for predicted observations. Building on this derivation approach, a computationally tractable CPHD filter for superpositional sensors was derived by Mahler and El-Fallah in [38].

Multi-Bernoulli filters

Multi-Bernoulli filters have been developed for the related pixelized image observation model [142]. Under the pixelized image observation model the likelihood has a separable form. This assumption is valid when the objects are non-overlapping. The developed filter equations can be used for track-before-detect applications without the need for measurement preprocessing. Hoseinnezhad et al. [116, 117] have used this filter for tracking multiple targets in image sequences and for audio-visual tracking in [144]. Since measurements corresponding to different targets are not allowed to overlap in the pixelized image observation model, these filters cannot be directly applied to superpositional observations without further approximation. Superpositional observations from phased array of acoustic sensors [145, 146] are processed using this filter by approximating the likelihood function with a multiple signal classification algorithm [133] based pseudo-likelihood function.

The recently developed labelled multi-Bernoulli RFS and generalized labelled multi-Bernoulli RFS [88, 89] have been extended to perform multitarget tracking using non-standard measurement models [141, 147, 148]. In [147], Papi and Kim have proposed a particle based multitarget tracking algorithm for superpositional measurements. They used the approximate CPHD filter [24] update equations to design proposal distributions for propagating labelled random finite sets over time. Labelled RFS based tracking has also been applied for the case of merged measurements [141] where groups of closely spaced targets can generate a single merged measurement. An approximation mechanism is developed in [148] for multitarget tracking using labelled random finite sets applicable to any generic measurement model.

2.11 Conclusions

In this chapter we introduced the relevant background material required for further reading of this thesis. We also provided an extensive literature review which highlights some of the limitations of the existing research work. For example, passive RF tomographic tracking has mostly been applied for single target tracking and in outdoor environments. In Chapter 3 we demonstrate passive multitarget RF tomographic tracking in both outdoor and relatively challenging indoor environments.

Traditional methods of processing superpositional measurements either require stationarity assumptions or depend on approximations which can lead to loss of information. To accurately perform sequential state estimation in a computationally tractable manner from superpositional measurements , we develop various random finite set based filters and their implementations in Chapter 4 and Chapter 5.

Although numerous filters and their implementations have been developed for standard sensors, they are largely restricted to the case when a single sensor is making observations. The existing multisensor filters suffer from drawbacks such as dependence on sensor order, numerical instability, and high computational requirements. To address these limitations, in Chapter 6 we propose a new multisensor filter and its greedy implementation which is computationally feasible, stable, and has no significant dependance on the order in which sensors are processed.

Chapter 3

Radio frequency tomography

In "device-free" multitarget tracking, targets do not carry active or passive devices to assist the tracking system. Applications of such a tracking system can be found in military surveillance, search-and-rescue operations, through-the-wall imaging, and healthcare environments [53, 58]. In the past couple of decades radio frequency (RF) measurements have been increasingly used for localization and target tracking. Wireless sensor networks that perform tracking using radio frequency sensors are attractive because

(a) radio frequency sensors are relatively inexpensive;

(b) sensor networks can be deployed easily and quickly;

(c) compared to the other alternatives such as infrared and video, RF measurements have the advantage that they can penetrate walls and other non-metallic obstacles.

Radio frequency tomography is the process of monitoring an area to detect mobile targets based on the additional attenuation and fluctuations they cause in wireless transmissions [5]. When a RF sensor network communicates through transmission of wireless packets, the *received signal strength* (RSS) measurements on a link connecting sensor nodes is affected by the distance between the sensors and the interference caused by static and moving objects. When multiple links monitor a region, the relative RSS measurements among the different links can be used to localize moving objects. The main contributions of this chapter are to propose and experimentally validate models for RSS measurements on RF links in presence of multiple targets and to demonstrate multitarget tracking using field deployments of RF sensor networks in both outdoor and indoor environments.

This chapter is organized as follows: Section 3.1 describes a general radio frequency sensor network setup. The empirical measurement models used to model the single-link observations in the presence of multiple targets are discussed in Section 3.2. The multitarget tracking algorithms used for processing sensor network data are discussed in Section 3.3. The data collection process and tracking results from outdoor and indoor deployment of radio frequency sensor networks are studied in Sections 3.4 and 3.5 respectively. Conclusions are presented in Section 3.6.

3.1 Radio frequency sensor network

Radio frequency sensor nodes are distributed over the periphery of the region of interest. Figure 3.1 shows an example of such a network with 20 nodes. The periphery can in general have any shape and is not limited to a square. Each node successively broadcasts packets at short time intervals and the other nodes measure the RSS. Each node pair constitutes a bidirectional link and has two measurements, with one on each of the forward and reverse links. The average of these two RSS measurement values is associated with the bidirectional link. Links which include either sensor 1 or 12 are indicated by dashed lines in Figure 3.1. For a wireless sensor network with R nodes there are $n_{\mathbf{z}} = \frac{R(R-1)}{2}$ bidirectional links. The measurements are then the RSS values of all bidirectional links stacked into a single measurement vector $\boldsymbol{\gamma}$ of length $n_{\mathbf{z}}$. All the nodes in the network transmit their local RSS measurements to a central receiving node which processes the data centrally. A single measurement interval corresponds to the period required for all nodes to transmit.

Let the RSS value recorded on link j be denoted by γ_j. The RSS value γ_j can be split into three main terms: $\gamma_j = \bar{\gamma}_j + y_j + w_j$. Here $\bar{\gamma}_j$ is the average RSS on link j when no target is present, y_j is the change in RSS value on link j due to mobile targets, and w_j is the noise term affecting the measured RSS on link j. We assume that there is a time period during which we can gather measurements on all links when no target is present in order to estimate $\bar{\gamma}_j$.

3.2 Radio frequency measurement models

In this section we characterize the changes in received signal strength measurements for a single RF link in presence of targets. We validate the proposed models using experimental

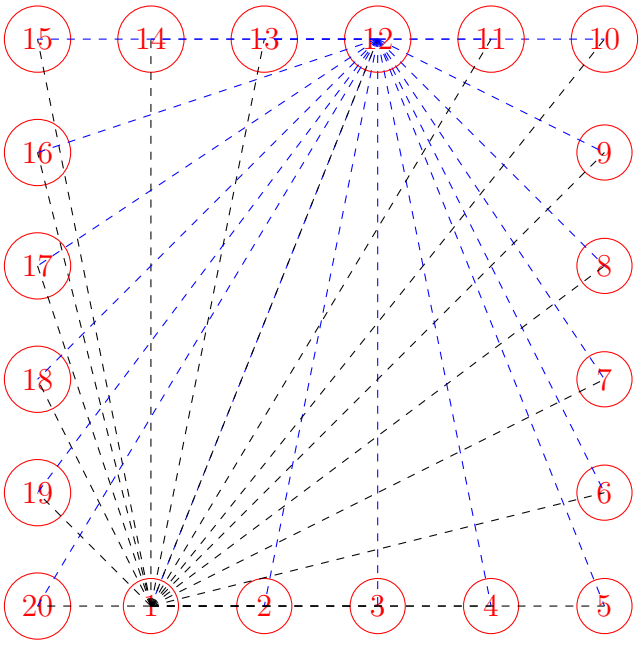


Fig. 3.1 Radio frequency sensor network: RF links which include either sensor 1 or 12 are indicated by dashed lines.

data.

3.2.1 Exponential model

A model for RSS attenuation on a single link in presence of a single target was proposed by Li et al. in [7]. This model was based on experimental data from relatively uncluttered outdoor regions. Let $\mathbf{x}$ be the position of the target affecting the link. The mean of the attenuation on the link caused by the target at position $\mathbf{x}$ is modeled as

$$g(\mathbf{x}) = \phi \exp\left(-\frac{\lambda(\mathbf{x})}{\sigma_\lambda}\right) \tag{3.1}$$

where ϕ and σ_λ are attenuation parameters based on physical properties of the targets and sensors and $\lambda(\mathbf{x})$ is defined as

$$\lambda(\mathbf{x}) = d_1(\mathbf{x}) + d_2(\mathbf{x}) - d_{12} \tag{3.2}$$

where $d_1(\mathbf{x})$ and $d_2(\mathbf{x})$ are the distances between the target and the two sensors and d_{12} is the distance between the sensors. $\lambda(\mathbf{x})$ captures the notion of the distance between the

target and the line-of-sight link between transmitter and receiver. Figure 3.2 graphically demonstrates the elliptical distance measure $\lambda(\mathbf{x})$. The ellipse corresponds to the collection of points $\mathbf{x}$ such that $\lambda(\mathbf{x})$ is constant.

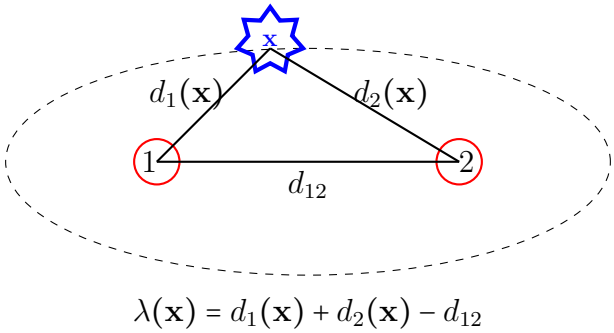


Fig. 3.2 The elliptical distance measure $\lambda(\mathbf{x})$ between the target at location $\mathbf{x}$ and the link formed by the sensors 1 and 2.

Let $g_j(\mathbf{x})$ be the RSS attenuation model for j^{th} link and w_j be the zero-mean Gaussian noise affecting the link. Then the observed reduction in RSS value $z_j = \gamma_j - \bar{\gamma}_j$ on link j due to target at position $\mathbf{x}$ is modeled as

$$z_j = g_j(\mathbf{x}) + w_j \tag{3.3}$$

$$w_j \sim \mathcal{N}(0, \sigma_{w_j}^2). \tag{3.4}$$

If there are $n_\mathbf{z}$ unique links the observed noisy attenuation is modeled as $\mathbf{z} = g(\mathbf{x}) + \mathbf{w}$, where $g = [g_1, g_2, \cdots, g_{n_\mathbf{z}}]$, and $\mathbf{w}$ is the additive zero-mean noise $\mathbf{w} \sim \mathcal{N}(0, \Sigma_\mathbf{w})$.

Thouin et al. proposed a multi-target extension of this model in [4]. The mean attenuation caused by the presence of multiple targets is modeled as equal to the sum of the mean attenuations due to each of the targets. Let set X be the collection of position of all the targets present. Then the mean of the total attenuation on link j due to all of the targets combined is modeled as:

$$\zeta_j(X) = \sum_{\mathbf{x} \epsilon X} g_j(\mathbf{x}) \tag{3.5}$$

This model has the superpositional form discussed in the Section 2.4.2. The observed noisy

attenuation vector is given by

$$\mathbf{z} = \zeta(X) + \mathbf{w} \tag{3.6}$$

where $\zeta = [\zeta_1, \zeta_2, \cdots, \zeta_{n_\mathbf{z}}]$ and $\mathbf{w}$ is the additive zero-mean noise $\mathbf{w} \sim \mathcal{N}(0, \Sigma_\mathbf{w})$. In the next section we provide experimental support for this model.

3.2.2 Experimental validation: Outdoor data

We assess the validity of the additive attenuation model using outdoor single-link sensor measurements when multiple targets affect the link. Our experiments are conducted for a link with the sensors separated by 8 meters. The transceivers of the sensor nodes are system-on-chip (SoC) TI CC2530 devices; each node has a monopole antenna and uses the 2.4 GHz IEEE 802.15.4 standard for communication. Figure 3.7 depicts the single link outdoor experimental setup.

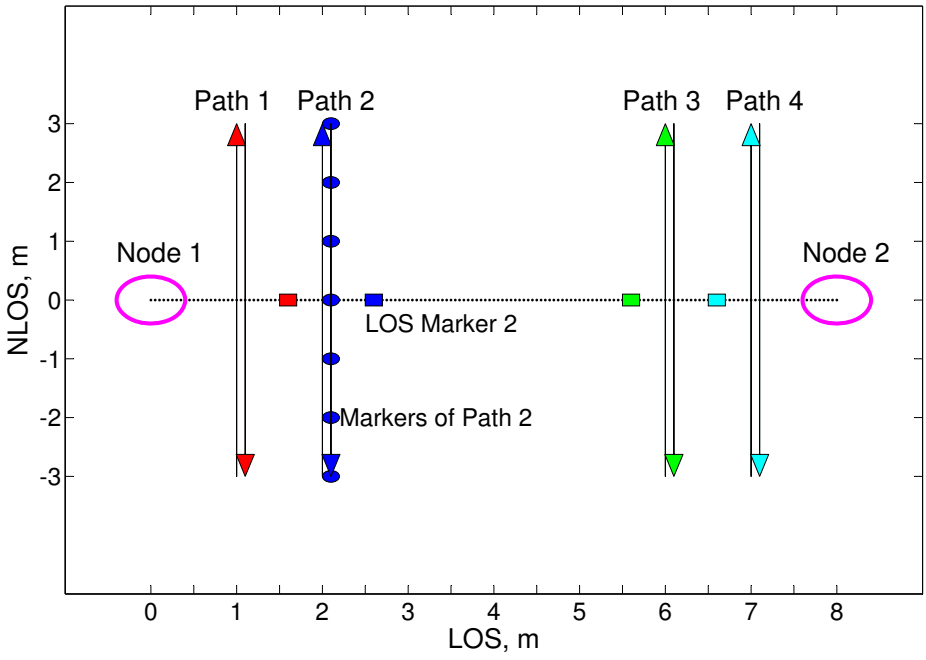


Fig. 3.3 The figure illustrates the single link experiment setup. Markers on LOS indicate different positions of stationary Target A. Arrows indicate path of moving Target B.

In each experiment, Target A stands on the line-of-sight path between the sensor nodes

(the positions are labeled "LOS marker" in Figure 3.7). Target B walks along a trajectory perpendicular to the line-of-sight path, crossing close to the position of Target A. For each experiment, Target B completes 10 crossings and approximately 22,000 RSS measurements are measured by the receiving sensors. We also conduct similar experiments when only Target A or Target B is present.

Figure 3.4 presents a scatter plot of the attenuation values after the background RSS estimates have been subtracted. This figure also shows the average attenuations when Target A is alone (dashed line) and when Target B is alone (blue line with square markers). We compare the average attenuation of the two targets with the model in (3.1). In the model we set ϕ for each target to the mean attenuation value when the target is alone and choose $\sigma_\lambda = 0.04$, which has been observed to be a good fit for the outdoor environment [8]. The additive model provides a good explanation of the experimental average attenuation, particularly in the region of interest (small λ).

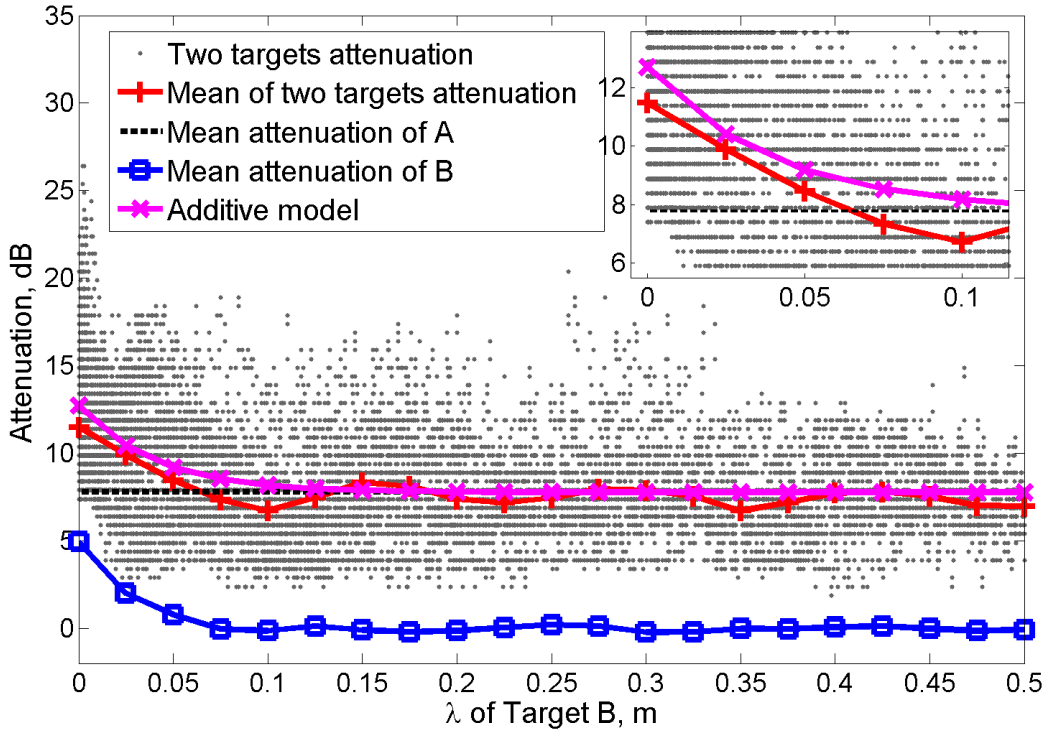


Fig. 3.4 Signal attenuation levels versus parameter λ for the single target and two targets case. Both experimental and model predicted attenuation levels are plotted when two targets are present.

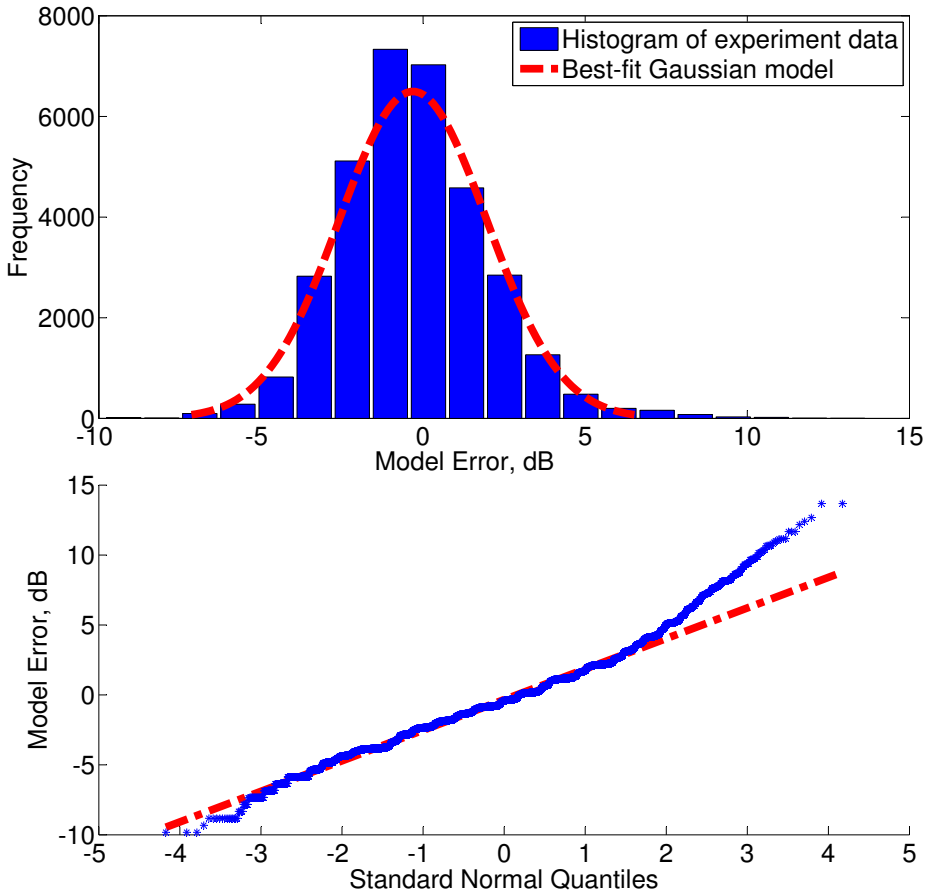


Fig. 3.5 Validation of the error as Gaussian model. Top figure shows a histogram of the model error and the corresponding best-fit Gaussian. The bottom figure shows a quantile-quantile plot of the model error.

Figure 3.5 depicts a histogram of the differences between the attenuation values predicted by the model (3.5) and the measured values. These differences should be explained by the additive Gaussian noise in (3.6). Superimposed is the best-fit Gaussian model. The Gaussian fit has mean $-0.029dB$ and standard deviation is $2.25dB$. The fact that the mean is close to zero is encouraging, but the histogram does not correspond to a Gaussian distribution. The quantile-quantile plot in Figure 3.5 shows significant discrepancy between the sample quantiles and the standard normal quantiles. There are multiple measurements where the observed attenuation is much higher than predicted by the model; the empirical distribution is skewed to the right and has a relatively heavy tail. The poor match is confirmed by a Lilliefors test, Anderson-Darling test, and D'Agostino test, which all reject

the null hypothesis that the differences are normally distributed (with significance 0.05).

Despite the poor fit of the Gaussian, we observe that using the Gaussian model still leads to reasonably good tracking performance (see Section 3.4). It simplifies the computations and since there are many hundreds of links it is important that log-likelihoods can be evaluated relatively quickly.

3.2.3 Magnitude model

In an outdoor environment, a radio frequency link usually experiences attenuation when a target is nearby. However, due to the multi-path effects caused by reflections from walls, ceilings, furniture, etc. in the indoor environment, a link can experience either attenuation or amplification when people move nearby. The single link measurement model outlined in Section 3.2.1 which models the attenuation of RSS values does not capture amplification effectively and performs poorly when used in an indoor setting.

To address this, we choose to model the magnitude of change in RSS values on link j, $z_j = |\gamma_j - \bar{\gamma}_j|$ using the model:

$$z_j = g_j(\mathbf{x}) + w_j \tag{3.7}$$

$$Prob(z_j|\mathbf{x}) \propto \mathcal{N}(g_j(\mathbf{x}), \sigma^2_{w_j}) \quad \text{for} \quad z_j > 0 \tag{3.8}$$

Here $g_j(\mathbf{x}_k)$ has the same form as in (3.1), but the parameter values are generally significantly different for indoor and outdoor environments. In above formulation the noise term w_j has a truncated Gaussian density (normalized to one) as given by (3.8).

Similar to (3.5), we propose a multi-target extension to the magnitude model. The magnitude of the change in RSS z_j is modeled as:

$$z_j = \zeta_j(X) + w_j \tag{3.9}$$

$$Prob(z_j|X) \propto \mathcal{N}(\zeta_j(X), \sigma^2_{w_j}) \quad \text{for} \quad z_j > 0 \tag{3.10}$$

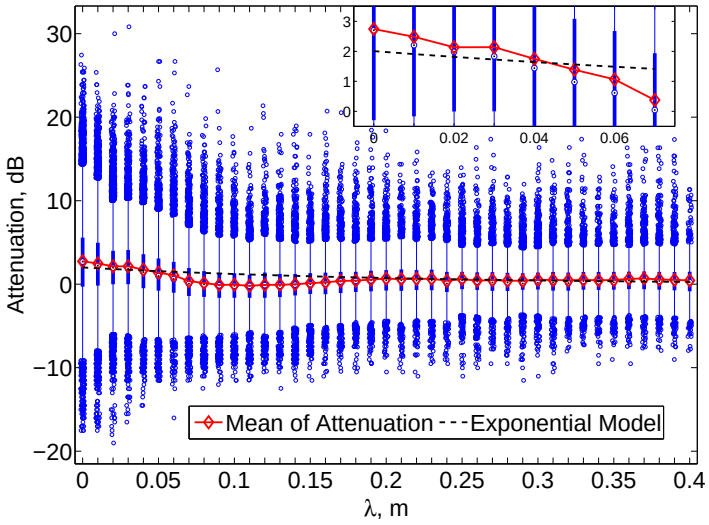
where $\zeta_j(X)$ has the superpositional form given in (3.5). Thus the observed magnitude of change in RSS is modeled as the sum of predicted changes in RSS caused by the individual targets and corrupted by truncated Gaussian noise.

3.2.4 Experimental validation: Indoor data

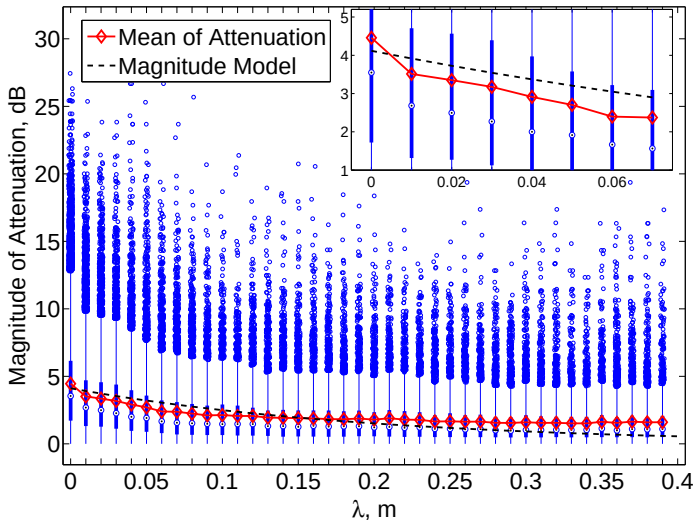
We first provide justification for the magnitude model for indoor data. Figures 3.6(a) and 3.6(b) display box-and-whisker plots of the RSS attenuation and magnitude of change in RSS respectively, for single target data recorded from Experiment 1 (see Section 3.5). The data are binned according to the λ values; for each bin, the central box ranges from the 25th to 75th percentile, the circle within the box indicates the median value, and the circles beyond the whiskers indicate outliers. Overlaid are the exponential and magnitude models, with parameters fitted using linear regression.

Both models achieve a relatively good fit to the means when λ is small, which is the important region. However, the magnitude model has a higher slope (see inset figures) and can thus more easily discern when the target is close to a link. For human targets, we have observed that the best-fit model parameters are similar for multiple indoor environments (with ϕ ranging from 3−7, and σ_λ ranging from 0.2−0.4). Tracking performance is relatively robust to the choice of these parameters. Figure 3.6(b) illustrates that the magnitude of change in RSS measurements are very noisy, with numerous outliers and heavy tails. The Gaussian noise model does not capture the tails particularly accurately but it is sufficient for tracking purposes and is more computationally tractable.

We now provide experimental support for the additive model when multiple targets are affecting the indoor RF link. Data was collected from a single indoor link of 7 meters (a representative link from Setup 2 in Section 3.5), comparing the cases where one target or two targets obstruct the link along its line-of-sight. Target A stood at different locations along the link and we recorded approximately 15,000 measurements of the resultant changes in RSS values. The procedure was repeated for target B at slightly different locations. We then made measurements with both A and B present at different combinations of the locations. The observations are depicted in Figure 3.7. The mean magnitude of change in RSS values are 2.78 and 2.74 for the individual targets, and 4.62 for the two targets. The distribution of attenuation magnitude for a combination of targets has a significantly heavier tail; a more sophisticated model could strive to capture this effect in addition to the increased mean.



(a) Indoor data with exponential model



(b) Indoor data with the magnitude model

Fig. 3.6 Box plots of the attenuation (top) and its magnitude (bottom) from Experiment 1 (see Section 3.5). Overlaid are the exponential and magnitude models.

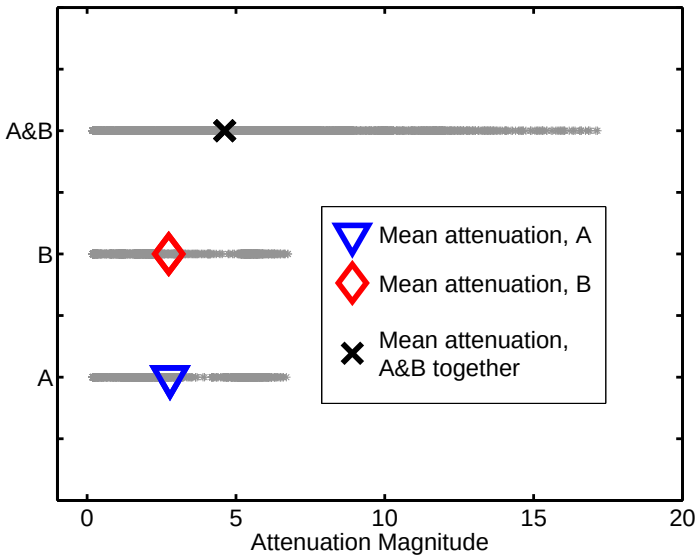


Fig. 3.7 Magnitude of RSS attenuation when one or two targets are present along the line-of-sight of a single link; experiments were conducted for a link in Setup 2 (see Section 3.5). Scattered points depict attenuation values averaged over short intervals and the markers indicate the mean values.

3.2.5 Skew-Laplace model

In [60], Wilson et al. proposed the use of the skew-Laplace distribution to model RSS attenuation $z_j = \gamma_j - \bar{\gamma}_j$ on link j. The skew-Laplace likelihood is defined as:

$$p(z_j|\mathbf{x}) = p(z_j|\mathbf{x}; a, b, \psi) \tag{3.11}$$

$$= \begin{cases} \frac{ab}{a+b} e^{-a(\psi - z_j)}, & if\ z_j \leq \psi \\ \frac{ab}{a+b} e^{-b(z_j - \psi)}, & otherwise \end{cases} \tag{3.12}$$

Here a and b represent the one-sided decay rates of the distribution for values less than or greater than the mode ψ. The parameters a, b and ψ are modeled as linearly dependent on the "fade level(F)" [60] for each link. The fade level quantifies the amount of fading when no targets are present and is estimated using measurements performed during a training period.

We conducted experiments in three different indoor locations (see Section 3.5) and collected more than three million data points. Data is collected when a single target is

moving inside the network and classified into training and testing data. Each experiment is repeated multiple (8-10) times and half of the repetitions are used for training models. Training data from all the experiments is combined and used to obtain model parameters. Table 3.1 details the linear fit parameters for the quantities a, b and ψ obtained using this training data. The linear fit models depend on whether the target is present on the line of sight (LOS) or away from it, non line of sight (NLOS).

	Parameter		
Target location	a	b	ψ
LOS	-0.0005F+0.31	-0.008F+0.63	-0.19F-0.69
NLOS	0.09F+1.81	0.14F+2.59	-0.01F+0.09

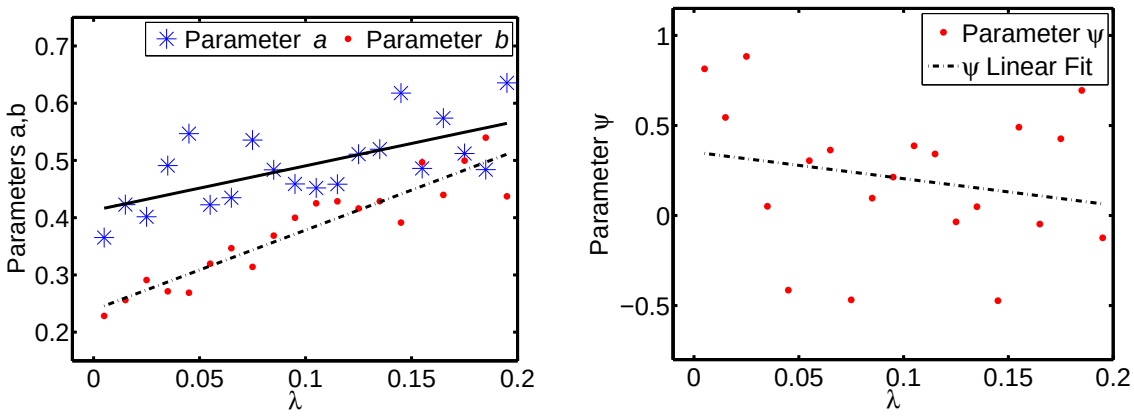
Table 3.1 Parameter values for fitted skew-Laplace distributions as linear function of fade level (F).

We observed that the RSS attenuation distributions vary as the value of the elliptical distance λ changes. Modifying the model proposed in [60], we model the parameters a, b, and ψ as linear functions of λ. For $\lambda < 0.2$, we divide the attenuation measurements into bins of width 0.01 in terms of λ; conduct a grid-search to identify the best-fit skew-Laplace parameters for each bin; and perform linear regression on these best-fit parameters to obtain linear models. Figures 3.8(a) and 3.8(b) and Table 3.2 show the fits we obtain. For $\lambda > 0.2$, the target location has little impact on the RSS attenuation and we model the parameters as constant.

	Parameter		
Target λ	a	b	ψ
$0 < \lambda \leq 0.2$	0.78λ+0.41	1.40λ+0.24	-1.47λ+0.35
$\lambda > 0.2$	1.29	1.08	-0.01

Table 3.2 Parameter values for fitted skew-Laplace distributions.

The upward trends of parameters a and b indicate that the distribution tends to become more peaky as λ increases (there are fewer large-magnitude attenuations/amplifications). For the parameter ψ the trend is less noticeable, but exhibits a downward trend, reflecting our observation that amplifications become rarer as λ increases. In later measurement based simulations we will compare the original skew-Laplace model [60] and the modified model proposed in this section.



(a) Parameter a, b.

(b) Parameter ψ

Fig. 3.8 Linear fits to the skew-Laplace parameters a and b obtained using training data from all three single-target experiments described in Section 3.5. The $\star$ and $\cdot$ markers show the best fit values for each λ bin.

3.3 Multitarget tracking algorithms

As seen from the previous section the measurement models in radio frequency tomography are highly non-linear functions of the multitarget state. Algorithms such as the extended or unscented Kalman filter do not perform well with these models. Hence we apply particle-based multitarget tracking algorithms for radio frequency tomographic tracking.

Let there be N fixed and known number of targets. The multitarget state at time step k is then represented by the vector $\mathbf{X}_k = [\mathbf{x}_{k,1}, \mathbf{x}_{k,2}, \ldots \mathbf{x}_{k,N}]$ where $\mathbf{x}_{k,j}$ is the $n_{\mathbf{x}}$ dimensional vector describing the state of the j^{th} target. The individual targets evolve independent of one another and let $t_{k|k-1}(\mathbf{x}_{k,j}|\mathbf{x}_{k-1,j})$ be the single target transition probability density. The measurement vector at time step k is denoted $\mathbf{z}_k$ and the collection of measurement vectors up until time step k is denoted by $\mathbf{z}_{1:k} = [\mathbf{z}_1, \mathbf{z}_1, \ldots, \mathbf{z}_k]$.

3.3.1 Sequential Importance Resampling filter

The first filter we discuss is based on the principle of sequential importance resampling (SIR) and is commonly known as the bootstrap particle filter [1] or the sequential Monte Carlo filter. It approximates the multitarget posterior distribution $f(\mathbf{X}_k|\mathbf{z}_{1:k})$ at each time

step k using a weighted set of particles:

$$f(\mathbf{X}_k|\mathbf{z}_{1:k}) \approx \sum_{i=1}^{N_p} w_k^{(i)} \delta(\mathbf{X}_k^{(i)}) \tag{3.13}$$

$$\sum_{i=1}^{N_p} w_k^{(i)} = 1 \tag{3.14}$$

where $\delta(\mathbf{X})$ is the Dirac delta function. A pseudo code for the SIR filter is given in Figure 3.9.

Initialize N particle filters $\{w_0^{(i)}, \mathbf{X}_0^{(i)}\}_{i=1}^{i=N_p}$
for $k = 1$ to T **do**
    for $i = 1$ to N_p **do**
        for $n = 1$ to N **do**
            proposal: $\mathbf{x}_{k,n}^{(i)} \sim t_{k|k-1}(\mathbf{x}_{k,n}|\mathbf{x}_{k-1,n}^{(i)})$
        end for
        weight update: $w_k^{(i)} \propto w_{k-1}^{(i)} \, L(\mathbf{z}_k|\mathbf{X}_k^{(i)})$
    end for
    resample step: $\{w_k^{(i)}, \mathbf{X}_k^{(i)}\}_{i=1}^{i=N_p} \rightarrow \{\frac{1}{N_p}, \mathbf{X}_k^{(i)}\}_{i=1}^{i=N_p}$
    state estimation: $\hat{\mathbf{X}}_k = \frac{1}{N_p} \sum_{i=1}^{N_p} \mathbf{X}_k^{(i)}$
end for

Fig. 3.9 Sequential Importance Resampling (SIR) filter [1].

The filter runs for T steps, processing one measurement vector at each time step. The filter prediction step consists of generating samples from the prior distribution. In the filter update step the weight of the particles is multiplied by the multitarget likelihood $L(\mathbf{z}_k|\mathbf{X}_k)$. Resampling (drawing a new set of unweighted particles from the weighted set) is performed periodically to maintain diversity of the particle set. The multitarget state estimate is obtained by performing a weighted average of the current particle set. We set $N_p = N \times N_{ppt}$ where N_{ppt} is the number of particles per target. The SIR filter operates in the $N \times n_{\mathbf{x}}$ dimensional space.

3.3.2 Multiple particle filter

The SIR filter can perform poorly when the state has high dimension, which occurs with multiple targets. To address this, Bugallo et al. proposed the multiple particle filter

(MPF) [2], which uses one low-dimensional particle filter for each target. The pseudo code of the multiple particle filter is provided in Figure 3.10.

Initialize N particle filters $\{w_{0,n}^{(i)}, \mathbf{x}_{0,n}^{(i)}\}_{i=1}^{i=N_{ppt}}$
for $k = 1$ to T **do**
    for $i = 1$ to N_{ppt} **do**
        for $n = 1$ to N **do**
            proposal: $\mathbf{x}_{k,n}^{(i)} \sim t_{k|k-1}(\mathbf{x}_{k,n}|\mathbf{x}_{k-1,n}^{(i)})$
        end for
    end for
    for $n = 1$ to N **do**
        predictive state estimate: $\tilde{\mathbf{x}}_{k,n} = \sum_{i=1}^{i=N_p} w_{k-1,n}^{(i)} \mathbf{x}_{k,n}^{(i)}$
    end for
    for $n = 1$ to N **do**
        for $i = 1$ to N_{ppt} **do**
            weight update: $w_{k,n}^{(i)} \propto w_{k-1,n}^{(i)} L(\mathbf{z}_k|\mathbf{x}_{k,n}^{(i)}, \tilde{\mathbf{X}}_{k,\underline{n}})$
        end for
        resample step: $\{w_{k,n}^{(i)}, \mathbf{x}_{k,n}^{(i)}\}_{i=1}^{i=N_{ppt}} \rightarrow \{\frac{1}{N_{ppt}}, \mathbf{x}_{k,n}^{(i)}\}_{i=1}^{i=N_{ppt}}$
        state estimation: $\hat{\mathbf{x}}_{k,n} = \frac{1}{N_{ppt}} \sum_{i=1}^{N_{ppt}} \mathbf{x}_{k,n}^{(i)}$
    end for
end for

Fig. 3.10 Multiple Particle Filter (MPF) [2].

For each of the N targets the MPF maintains a separate particle filter using N_{ppt} particles. We implement each of these as an SIR filter. The weight update step for the individual filters cannot be performed independently of the other target states because computing the measurement likelihood requires the combined state information. The MPF uses an estimate for the other target states, $\tilde{\mathbf{X}}_{k,\underline{n}} = [\tilde{\mathbf{x}}_{k,1} \ldots \tilde{\mathbf{x}}_{k,n-1}, \tilde{\mathbf{x}}_{k,n+1} \ldots \tilde{\mathbf{x}}_{k,N}]$, where $\tilde{\mathbf{x}}_{k,j}$ is the predictive state estimate of the j^{th} target obtained by performing a weighted average of the current particles $\{\mathbf{x}_{k,j}^{(i)}\}$ using the weights from the previous time step $\{w_{k-1,j}^{(i)}\}$. Each of the individual particle filters operate in the $n_{\mathbf{x}}$ dimensional space.

3.3.3 MCMC filter

The third algorithm we discuss is the Markov chain Monte Carlo (MCMC) filter of [3]. Samples (particles) are drawn at each time step by sequentially traversing a Markov chain which has the desired density function as its stationary distribution. We use a sequential

implementation of the MCMC filter that approximates the posterior multitarget density using a set of (unweighted) particles. The MCMC filter pseudo code is given in Figure 3.11.

Initialize particles $\{\mathbf{X}_0^{(i)}\}_{i=1}^{i=N_p}$
for $k = 1$ to T **do**
    initialize MCMC chain: $(\mathbf{X}_k^{[0]}, \mathbf{X}_{k-1}^{[0]})$
    for $m = 1$ to $(N_{burn} + N_p \times N_{thin})$ **do**
        joint draw: $(\mathbf{X}_k^*, \mathbf{X}_{k-1}^*) \sim q_1(\mathbf{X}_k, \mathbf{X}_{k-1} | \mathbf{X}_k^{[m-1]}, \mathbf{X}_{k-1}^{[m-1]})$
        acceptance: $(\mathbf{X}_k^{[m]}, \mathbf{X}_{k-1}^{[m]}) = (\mathbf{X}_k^*, \mathbf{X}_{k-1}^*)$ with probability ρ_1
        for $n = 1$ to N **do**
            refinement: $\mathbf{x}_{k,n}^* \sim q_2(\mathbf{x}_{k,n} | \mathbf{X}_k^{[m]}, \mathbf{X}_{k-1}^{[m]})$
            acceptance: $\mathbf{x}_{k,n}^{[m]} = \mathbf{x}_{k,n}^*$ with probability ρ_2
        end for
    end for
    for $i = 1$ to N_p **do**
        selection: $\mathbf{X}_k^{(i)} = \mathbf{X}_k^{[N_{burn} + i \times N_{thin}]}$
    end for
end for

Fig. 3.11 MCMC Filter [3]

Let the posterior multitarget density at time $k-1$ be approximated using an unweighted set of particles as

$$f(\mathbf{X}_{k-1} | \mathbf{z}_{1:k-1}) \approx \frac{1}{N_p} \sum_{i=1}^{N_p} \mathbf{X}_{k-1}^{(i)}. \tag{3.15}$$

The firs step of the algorithm is to generate samples from the joint density function $f(\mathbf{X}_k, \mathbf{X}_{k-1} | \mathbf{z}_{1:k})$ which can be expressed as

$$f(\mathbf{X}_k, \mathbf{X}_{k-1} | \mathbf{z}_{1:k}) \propto L(\mathbf{z}_k | \mathbf{X}_k)\, t_{k|k-1}(\mathbf{X}_k | \mathbf{X}_{k-1})\, f(\mathbf{X}_{k-1} | \mathbf{z}_{1:k-1}) \tag{3.16}$$

The joint draw step samples the joint vector $(\mathbf{X}_k, \mathbf{X}_{k-1})$ from the proposal distribution $q_1(\mathbf{X}_k, \mathbf{X}_{k-1} | \mathbf{X}_k^{[m-1]}, \mathbf{X}_{k-1}^{[m-1]})$ using the Metropolis-Hastings algorithm [149]. The superscript $[m]$ notation denotes the m^{th} sample generated by the MCMC chain. The proposed sample

$(\mathbf{X}_k^*, \mathbf{X}_{k-1}^*)$ is accepted with probability ρ_1

$$\rho_1 = \min\left(1, \frac{f(\mathbf{X}_k^*, \mathbf{X}_{k-1}^*|\mathbf{z}_{1:k})}{q_1(\mathbf{X}_k^*, \mathbf{X}_{k-1}^*|\mathbf{X}_k^{[m-1]}, \mathbf{X}_{k-1}^{[m-1]})} \frac{q_1(\mathbf{X}_k^{[m-1]}, \mathbf{X}_{k-1}^{[m-1]}|\mathbf{X}_k^*, \mathbf{X}_{k-1}^*)}{f(\mathbf{X}_k^{[m-1]}, \mathbf{X}_{k-1}^{[m-1]}|\mathbf{z}_{1:k})}\right) \tag{3.17}$$

We choose the proposal density q_1 as:

$$q_1(\mathbf{X}_k, \mathbf{X}_{k-1}|\mathbf{X}_k^{[m-1]}, \mathbf{X}_{k-1}^{[m-1]}) \propto \sum_{i=1}^{N_p} t_{k|k-1}(\mathbf{X}_k|\mathbf{X}_{k-1}^{(i)})\delta(\mathbf{X}_{k-1}^{(i)}) \tag{3.18}$$

The refinement step uses Gibbs sampling [150] to sequentially refine the individual target states. The marginal distribution for the n^{th} target is given by $f(\mathbf{x}_{k,n}|\mathbf{X}_{k,\underline{n}}, \mathbf{X}_{k-1}, \mathbf{z}_{1:k})$ where $\mathbf{X}_{k,\underline{n}} = [\mathbf{x}_{k,1}, \ldots, \mathbf{x}_{k,n-1}, \mathbf{x}_{k,n+1}, \ldots, \mathbf{x}_{k,N}]$. If the sample $\mathbf{x}_{k,n}^*$ is generated using the proposal density $q_2(\mathbf{x}_{k,n}|\mathbf{X}_k^{[m]}, \mathbf{X}_{k-1}^{[m]})$, it is accepted with probability ρ_2 given by

$$\rho_2 = \min\left(1, \frac{f(\mathbf{x}_{k,n}^*|\mathbf{X}_{k,\underline{n}}^{[m]}, \mathbf{X}_{k-1}^{[m]}, \mathbf{z}_{1:k})}{q_2(\mathbf{x}_{k,n}^*|\mathbf{X}_k^{[m]}, \mathbf{X}_{k-1}^{[m]})} \frac{q_2(\mathbf{x}_{k,n}^{[m]}|\mathbf{x}_{k,n}^*, \mathbf{X}_{k,\underline{n}}^{[m]}, \mathbf{X}_{k-1}^{[m]})}{f(\mathbf{x}_{k,n}^{[m]}|\mathbf{X}_{k,\underline{n}}^{[m]}, \mathbf{X}_{k-1}^{[m]}, \mathbf{z}_{1:k})}\right) \tag{3.19}$$

We choose the proposal density q_2 as:

$$q_2(\mathbf{x}_{k,n}|\mathbf{X}_k^{[m]}, \mathbf{X}_{k-1}^{[m]}) = t_{k|k-1}(\mathbf{x}_{k,n}|\mathbf{x}_{k-1,n}^{[m]}) \tag{3.20}$$

At every time step, the MCMC chain is initialized with a particle from the previous time step that has the highest likelihood for the current observation. A burn-in of N_{burn} samples and a thinning factor of N_{thin} reduce the correlation between the consecutive samples.

3.3.4 PHD filter

The final filter we assess is the Probability hypothesis density (PHD) filter (see Section 4.2). The algorithms discussed earlier either try to sample from the multitarget posterior density (SIR and MCMC) or from the individual target marginal posterior densities (MPF), but the PHD filter samples from the first moment of the multitarget posterior density, also called the Probability Hypothesis Density (PHD) [86]. The PHD is a function over the single target state space. It is high in regions where targets are present and its integral over the target state space is equal to the expected number of targets. Thus sampling from the PHD populates particles in region where there is high probability of targets being

present. The advantage of the PHD filter is that we do not have to sample from the high dimensional space of multiple targets. Please see Section 4.2 for the update equations and a detailed discussion on the PHD filter.

Initialize particles $\{w_0^{(i)}, \mathbf{x}_0^{(i)}\}_{i=1}^{i=N_p}$
for $k = 1$ to T **do**
    for $i = 1$ to N_p **do**
        proposal: $\mathbf{x}_k^{(i)} \sim t(\mathbf{x}_k | \mathbf{x}_{k-1}^{(i)})$
    end for
    $\hat{\mu}_k = \sum_i w_{k-1}^{(i)} g(\mathbf{x}_k^{(i)})$
    $\hat{\Sigma}_k = \sum_i w_{k-1}^{(i)} g(\mathbf{x}_k^{(i)}) g(\mathbf{x}_k^{(i)})^T$
    for $i = 1$ to N_p **do**
        weight update: $w_k^{(i)} = w_{k-1}^{(i)} \frac{\mathcal{N}_{\mathbf{z}_k}(g(\mathbf{x}_k^{(i)}) + \hat{\mu}_k, \hat{\Sigma}_k + \Sigma)}{\mathcal{N}_{\mathbf{z}_k}(\hat{\mu}_k, \hat{\Sigma}_k + \Sigma)}$
    end for
    resample step: $\{w_k^{(i)}, \mathbf{x}_k^{(i)}\}_{i=1}^{i=N_p} \rightarrow \{\frac{1}{N_p}, \mathbf{x}_k^{(i)}\}_{i=1}^{i=N_p}$
    clustering: $\{\hat{\mathbf{x}}_{k,n}\}_{n=1}^{N} = \text{cluster}(\{\mathbf{x}_k^{(i)}\}_{i=1}^{i=N_p}, N)$
end for

Fig. 3.12 PHD Filter [4]

The pseudo code of the particle filter (SIR) based implementation of the PHD filter, assuming the number of targets is fixed and known, is given in Figure 3.12. The algorithm approximates the normalized PHD function using a set of weighted particles. The statistics $\hat{\mu}_k$ and $\hat{\Sigma}_k$ are estimated using the proposed particle set which are required in the weight update step. Resampling is performed to retain diversity of particles. We use the k-means clustering algorithm to group the particles into clusters, each of which correspond to a target. The centroids of these clusters are the target state estimates.

3.3.5 Algorithms for unknown target number

We now discuss an extension of the SIR and MCMC filters which allows them to address the case where the number of targets is unknown and varies with time. Following [71], we extend the single target state $\mathbf{x}_{k,n}$ to include an indicator variable $e_{k,n}$ which indicates the presence ($e_{k,n} = 1$) or absence ($e_{k,n} = 0$) of the target. In our analysis we assume that the existence indicator variable evolves independently of the target location and velocity and does so independently for each of the targets. This approach requires us to specify

the maximum number of targets $N_{\max}$ that can be present at any given time. The SIR and MCMC algorithms presented above are easily modified; there is just an additional propagation step for the indicator variable. Since the number of targets is unknown, we need to estimate the number of targets as well as the target locations. We employ a simple heuristic: at any time step we declare a target to be present if more than half of the corresponding target particles have their indicator variable set to 1.

3.4 Processing outdoor sensor network data

3.4.1 Experiments and data collection

We perform multiple experiments to gather sensor network data in an outdoor environment. Experiments are conducted using a network of 24 radio frequency sensor nodes deployed in a $7m \times 7m$ square layout in an empty grass field. The sensor network is graphically depicted in Figure 3.13. Transceivers of the sensor nodes are system-on-chip (SoC) TI CC2530 devices and they use 2.4 GHz IEEE 802.15.4 standard for communication. Sensor network deployment and collection of data was done by students at BUPT.

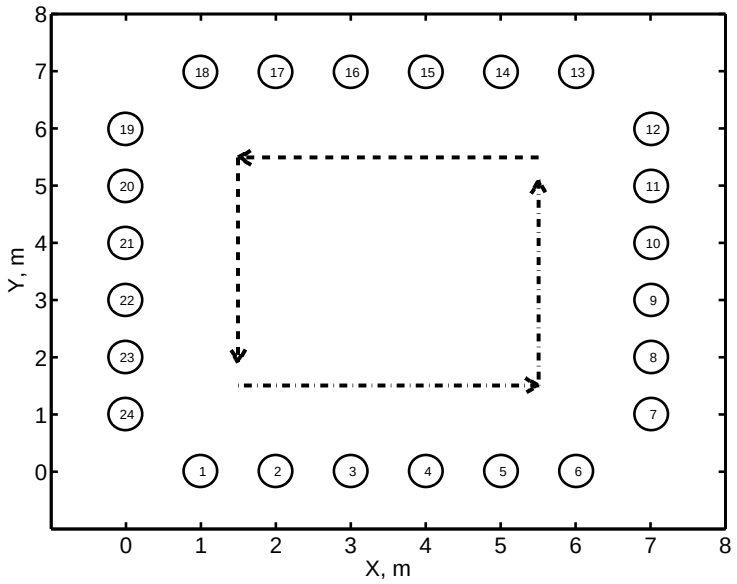


Fig. 3.13 Graphical representation of an outdoor sensor network consisting of 24 RF sensor nodes (indicated as numbered circles) deployed in a square layout and having $M = 276$ communication links. The dashed arrows indicate the direction and the path of the targets in Exp. 1.

A simple token ring protocol is used to control transmission. During each time interval of 8.3ms, one node broadcasts a data packet. All of the other nodes receive this packet and measure the RSS. The token is then passed to the next node. A complete set of measurements, corresponding to a transmission from each of the 24 sensors, is thus collected every 200ms. This constitutes one time step for the algorithms.

We perform three different experiments, repeating each ten times. In the first experiment (Exp. 1) two targets move inside the sensor network along a square trajectory. The targets start from diagonally opposite corners of the square and move in anti-clockwise direction as indicated by the arrows in Figure 3.13. In second experiment (Exp. 2) two targets move along a zigzag trajectory (see Figure 3.14(a)). In the third experiment (Exp. 3), four targets are present. Two targets move around a bigger square in an anti-clockwise direction, starting in opposite corners. The other two other targets move around a smaller square in a clockwise direction (see Figure 3.14(b)).

All of the targets are present for the entire duration of the experiment. Two rounds of the trajectory are completed by each target in experiments one and two. Two rounds of the outer square and four rounds of the inner square are completed by the targets in experiment three. Visible markers are placed along the target trajectories and we record the times when targets cross the marker locations to establish a ground-truth trajectory.

3.4.2 Simulation settings

We adopt a simple random walk model for the target motion. Our main motivation for the simple model is to allow for very general movements (humans walking, stopping, turning). The target state $\mathbf{x}$ is a 2-dim vector consisting of x and y coordinates of its position. The single target state evolution equation is given as:

$$\mathbf{x}_{k+1,n} = \mathbf{x}_{k,n} + \mathbf{v}_{k,n} \tag{3.21}$$

where $\mathbf{v}_{k,n}$ is the process noise distributed according to the Gaussian distribution $\mathbf{v}_{k,n} \sim \mathcal{N}(0, \sigma_v^2 I_{2\times 2})$. The targets are assumed to move independently of each other. We assume that σ_v is a known constant, but online expectation-maximization based procedures [7] could be incorporated to jointly track the targets and estimate the process noise variance. The process noise standard deviation is set to $\sigma_v = 0.2$ in all our simulations. This value effectively places a bound on the distance a target is likely to travel in 200ms (the mea-

surement interval) and the value 0.2 reflects that we are trying to track walking human targets.

We use the exponential model discussed in Section 3.2.1 for outdoor data processing. The measurement model parameters are set to be $\phi = 5$, $\sigma_\lambda = 0.04$. The sensor measurement noise covariance matrix is modeled as $\Sigma_{\mathbf{w}} = \sigma_w^2 I_{n_\mathbf{z} \times n_\mathbf{z}}$ with $\sigma_w = 1$. These values are selected because they are the average values observed in many outdoor experiments. The tracking performance is relatively robust to changes of these parameter values.

In the algorithms, we vary the number of particles per target over the range $N_{ppt} = 50, 250, 500$. For the MCMC algorithm, the burn-in is $N_{burn} = 1000$ and the thinning factor is $N_{thin} = 3$ which are standard values from the literature, observed to be sufficient in many cases to substantially reduce correlation between samples. For all of the filters, the initial set of particles for each target is independently drawn from a Gaussian distribution of unit variance, centered around the actual target location. The number of targets is assumed to be fixed and known by all the algorithms. The estimated target locations are obtained by computing the weighted average of the particles.

We use the OSPA metric (see Section 2.7) to measure the error between the estimated and the ground truth target locations. We use the value $p = 2$. The track estimates are obtained by connecting over time the best possible association at every time step.

All the processing was conducted on a cluster machine running MATLAB Distributed Computing Server. Distributed processing was used to speed up the simulations by distributing independent runs of the algorithm over different processors. The cluster machine consists of eight cluster nodes all with the same configuration. Each cluster node has 8 processors (two Xeon 4-core 2.5GHz processors) with a shared memory of 14GB RAM.

3.4.3 Tracking results

We run each algorithm with 10 different random initializations. Since each experiment is repeated 10 times, we compute an average error over all the 100 track-initialization pairs. Table 3.3 presents the average error for the three different experiments obtained by varying the number of particles per target as $N_{ppt} = 50$, 250, and 500. The table also shows the average computation time required per time step. Computational time is measured using the tic, toc routines in MATLAB.

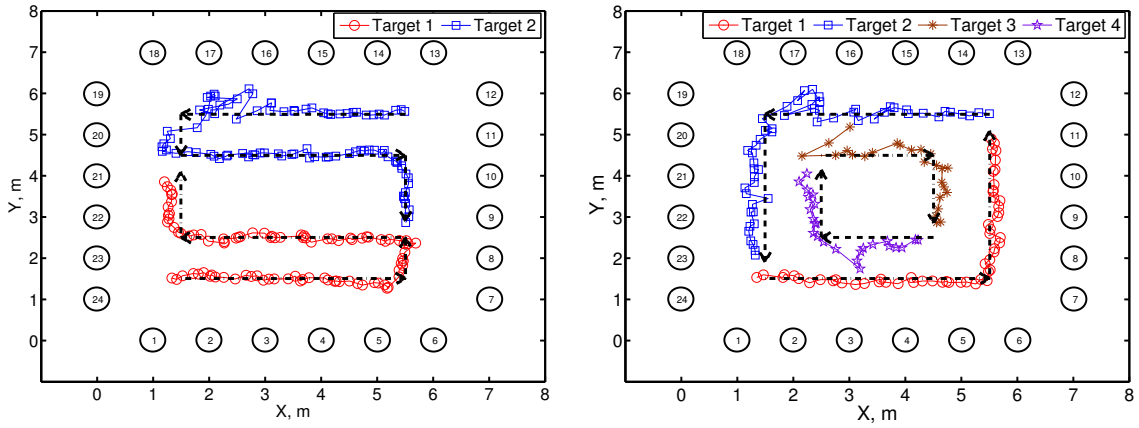
The results illustrate that the MPF requires only 50 particles per target to track ef-

		SIR		MPF		MCMC	
		Error	Time	Error	Time	Error	Time
Experiment	N_{ppt}	(m)	(ms)	(m)	(ms)	(m)	(ms)
Exp. 1 (Two targets)	50	0.42	20	0.20	17	0.23	1008
	250	0.21	101	0.20	88	0.20	1978
	500	0.20	217	0.20	179	0.20	3374
Exp. 2 (Two targets)	50	0.64	19	0.23	17	0.22	973
	250	0.23	95	0.22	84	0.22	1889
	500	0.27	202	0.22	170	0.22	3137
Exp. 3 (Four targets)	50	1.34	62	0.96	48	0.77	1709
	250	1.06	338	0.90	244	0.69	4333
	500	0.88	803	0.91	493	0.63	7636

Table 3.3 Average OSPA error (in meter) and average computational time (in milliseconds) per time step for different experiments using different algorithms with $N_{ppt} = 50, 250$ & 500.

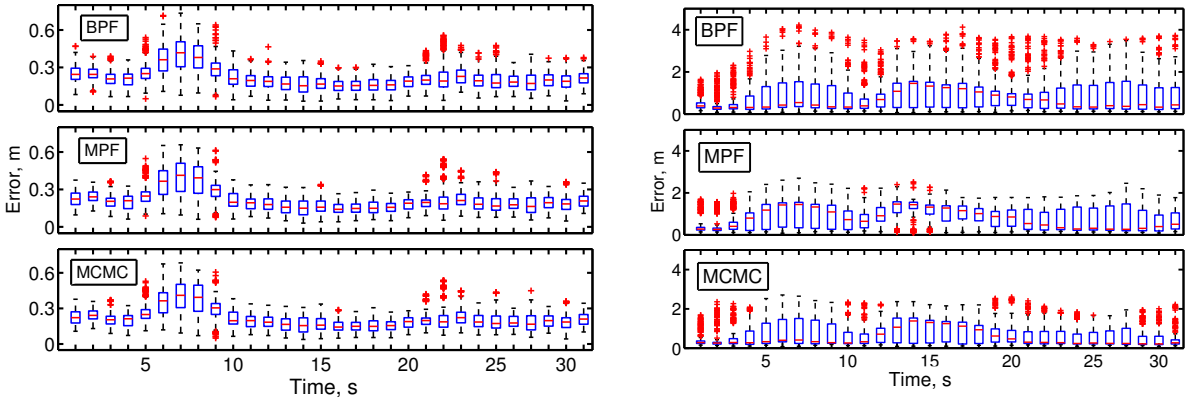
fectively and that it can execute in real-time (computational time is less than the 200ms measurement interval). It significantly outperforms the SIR, which needs many more particles to achieve comparable accuracy. For the two target experiments both MPF and MCMC have similar performance. For four targets, the MCMC filter is more accurate than the MPF, but it is almost impossible to execute as a real-time system. The results for the four-target experiment do illustrate that there is room for improvement over the MPF algorithm. Interestingly, the accuracy of the MPF does not improve significantly as the number of particles is increased, indicating that the extra error is probably due to the approximation employed in the algorithm during the weight update stage (see Section 3.3.2).

Sample tracking trajectories are shown in Figure 3.14(a) (two targets, MPF algorithm) and Figure 3.14(b) (four targets, MCMC filter). The tracking is generally accurate, with the occasional departure from the true trajectory. Figures 3.15(a) and 3.15(b) show box and whisker plots of error as a function of time for the different algorithms. The SIR has numerous outliers indicating frequent tracking errors. The MPF and MCMC filter have fewer outliers and the worst-case errors are on the order of 0.6m for the two-target case and 2.5m for the four-target case.



(a) Estimated tracks for Exp. 2 (b) Estimated tracks for Exp. 3

Fig. 3.14 The figures show the ground truth trajectory of two/four targets as indicated by the dashed arrows. (a) Exp. 2: The estimated target tracks obtained using the MPF algorithm with N_{ppt} = 500. (b) Exp. 3: The estimated target tracks obtained using the MCMC algorithm with N_{ppt} = 500.



(a) Box and whisker plot of error for Exp. 2 (b) Box and whisker plot of error for Exp. 3

Fig. 3.15 Box and whisker plot of the error for experiments 2 & 3 as simulation progresses for the algorithms SIR, MPF and MCMC with N_{ppt} = 500. The box ranges from 25^{th} to 75^{th} percentile and the line within the box indicates the median value and the pluses indicate outliers.

3.5 Processing indoor sensor network data

3.5.1 Experiments and data collection

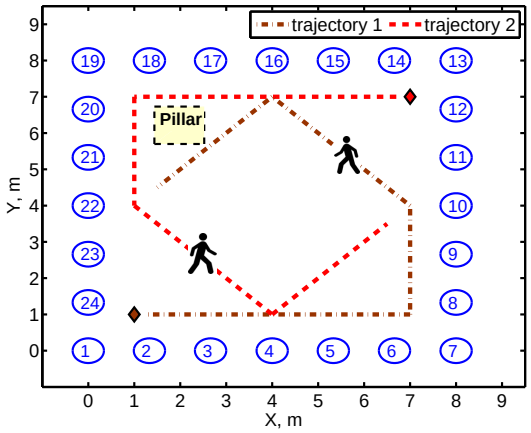
We performed multiple experiments at three different indoor sites and repeated each experiment multiple (8-10) times. The first site is in the Trottier Building at McGill Uni-

versity (Figure 3.16(b)). An area of $8m \times 8m$ was monitored by 24 RF sensor nodes (Figure 3.16(a)). It is referred to as Setup1 henceforth. A concrete pillar lies within the network. The second experiment site (Setup2) is the Computer Networks Lab of McGill University (Figure 3.16(d)). An area of $9m \times 9m$ is monitored by 24 sensor nodes (Figure 3.16(c)). Numerous desks and chairs are present within the network and there are walls just outside. The third experiment site (Setup3) is in the Beijing University of Posts and Telecommunications (BUPT), China. Data was collected in a completely through-wall environment (Figure 3.17(b)) using 28 nodes covering a $5.2m \times 6.7m$ region (Figure 3.17(a)). A high sensor density was used at this site because of the presence of walls within the sensor network. The same RF sensor nodes as described in Section 3.4 are used in all our setups and each time step corresponds to a measurement interval of 200ms.

Setup	Experiment		Description
Setup1	Exp. 1	single target	24 sensors in $8m \times 8m$, concrete pillar obstruction
	Exp. 4	two target	
	Exp. 6	three target	
Setup2	Exp. 2	single target	24 sensors in $9m \times 9m$, desks and chairs obstruction
	Exp. 5	two target	
	Exp. 7	time varying target number	
Setup3	Exp. 3	single target	28 sensors in $5.2m \times 6.7m$, through-wall environment

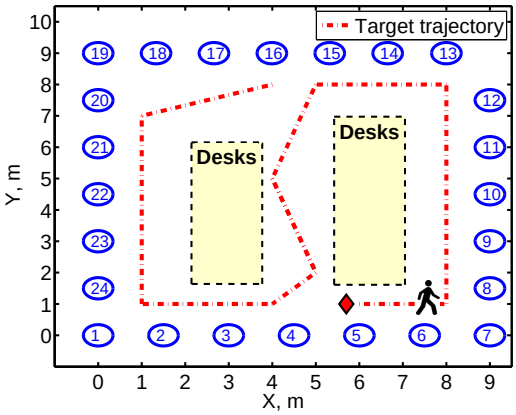
Table 3.4 Description of the different indoor experimental setups and the nature of the experiments used to collect data.

We performed multiple single and multi-target experiments to collect measurements from the sensor network. The setup description and experiments are summarized in Table 3.4. The single target experiments at Setup1, Setup2 and Setup3 are respectively referred to as "Exp. 1", "Exp. 2" and "Exp. 3". The two-target experiments at Setup1 and Setup2 are called "Exp. 4" and "Exp. 5" respectively. The three-target experiment performed at Setup1 is called "Exp. 6". The time-varying number of targets experiment at Setup2 is called "Exp. 7". In this experiment either one or two targets are present at any given instant.



(a) Setup1 layout.

(b) Photograph of Setup1.



(c) Setup2 layout.

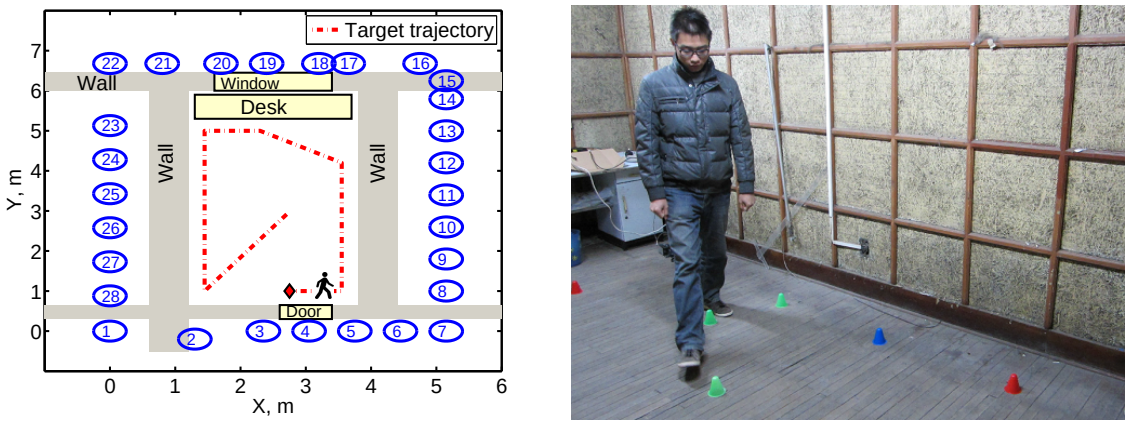
(d) Photograph of Setup2.

Fig. 3.16 Layouts and photos of experiments: (a), (b) Setup1, uncluttered indoor environment; (c), (d) Setup2, cluttered indoor environment.

3.5.2 Simulation settings

The target dynamics are modeled using a jump-state Markov model in our simulations. Jump-state Markov models are standard models for describing the dynamics of a maneuvering object [151, 152]. The model assumes that the target operates at each time step in one of multiple modes of operation (represented as a discrete state variable). The discrete state jumps/switches are independent of the target positions and are governed by a Markov chain.

We adopt the following jump-state Markov model in our experiments. A single target



(a) Setup3 layout.

(b) Photograph of Setup3.

Fig. 3.17 Layout and photo of Setup3, through-wall measurements.

state consists of its position $\mathbf{x}$ in the x-y plane, its current motion angle θ and a discrete state variable u. Let the initial state distribution at time step $k = 0$ is modeled as $p(u_0, \theta_0, \mathbf{x}_0)$. The target motion equations are then given as:

$$u_k \sim p(u_k|u_{k-1}), \tag{3.22}$$

$$\theta_k = \theta_{k-1} + \nu(u_k) + s_k, \tag{3.23}$$

$$\mathbf{x}_k = \mathbf{x}_{k-1} + m \begin{bmatrix} \cos\theta_k \\ \sin\theta_k \end{bmatrix} + \mathbf{v}_k. \tag{3.24}$$

Here the discrete state variable can take values $u_k \in \{0, 1, 2\}$ which represents "no turn" ($\nu(0) = 0$), "left-turn" ($\nu(1) = 0.1$ rad), and "right-turn" ($\nu(2) = -0.1$ rad), respectively. $p(u_k|u_{k-1})$ is the transition probability matrix and $s_k \sim \mathcal{N}(0, \sigma_s^2)$ and $\mathbf{v}_k \sim \mathcal{N}(0, \sigma_v^2 I_{2\times 2})$ are the innovation noise terms. The speed of the target is assumed to be constant m.

We set $m = 0.1$ in our experiments since it approximates the typical walking speed of humans. We model the transition probabilities $p(u_k|u_{k-1})$ as given in the table 3.5. The transition probabilities are chosen to approximate typical human motion characteristics. Higher values in the first row indicate a greater tendency to walk straight than to make turns. Angle innovation noise variance σ_s^2 is set to 0.001 to model smooth target motion. The tracking performance is robust to small changes in σ_s^2, m and the table entries. The parameter σ_v is set to 0.1, although we examine the impact of changing it to 0.2 for the

	u_{k-1}		
u_k	0	1	2
0	0.75	0.65	0.65
1	0.125	0.3	0.05
2	0.125	0.05	0.3

Table 3.5 Transition probabilities $p(u_k|u_{k-1})$ used in the simulations.

single-target case.

Observation model parameters were selected based on training phases performed at the three experimental sites. The test data sets do not include any data from the training phases. For the MCMC filter we use $N_{burn} = 1000$ and $N_{thin} = 3$. Some links exhibit large variance in their RSS measurements even when the network is vacant. Those links have severe impacts on the tracking results, as the variation of the RSS is not caused by the targets. Thus we exclude the RSS measurements of any links whose variance is higher than 1 for the vacant network as a pre-processing step. All the processing is done in MATLAB on the cluster machine described in Section 3.4.2.

3.5.3 Tracking results

To compare the algorithms we use the OSPA error metric when the number of targets are known and fixed and the OSPA error metric when the number of targets is unknown and changes over time. The reported tracking errors are calculated by averaging over the multiple repetitions of the experiments and for each experiment running the tracking algorithm with 10 different random initializations.

Single target tracking

The algorithms of SIR, MCMC and PHD are compared for the different single target experiments. The MPF is identical to the SIR algorithm in the single target case. For all filters, we set $N_p = 500$ (larger values were observed to give minimal improvement and smaller values lead to larger tracking error). Tracking is performed using the measurement models discussed in Section 3.2: the magnitude model (Mag), modified skew-Laplace model (SL) and original skew-Laplace model (SL [60]). The best-fit skew-Laplace parameters are given in Tables 3.1 and 3.2. For the magnitude model, we use $\sigma_w = 2$, $\phi = 4$ and $\sigma_\lambda = 0.2$. Since the particle implementation of the PHD filter in [4] is obtained assuming

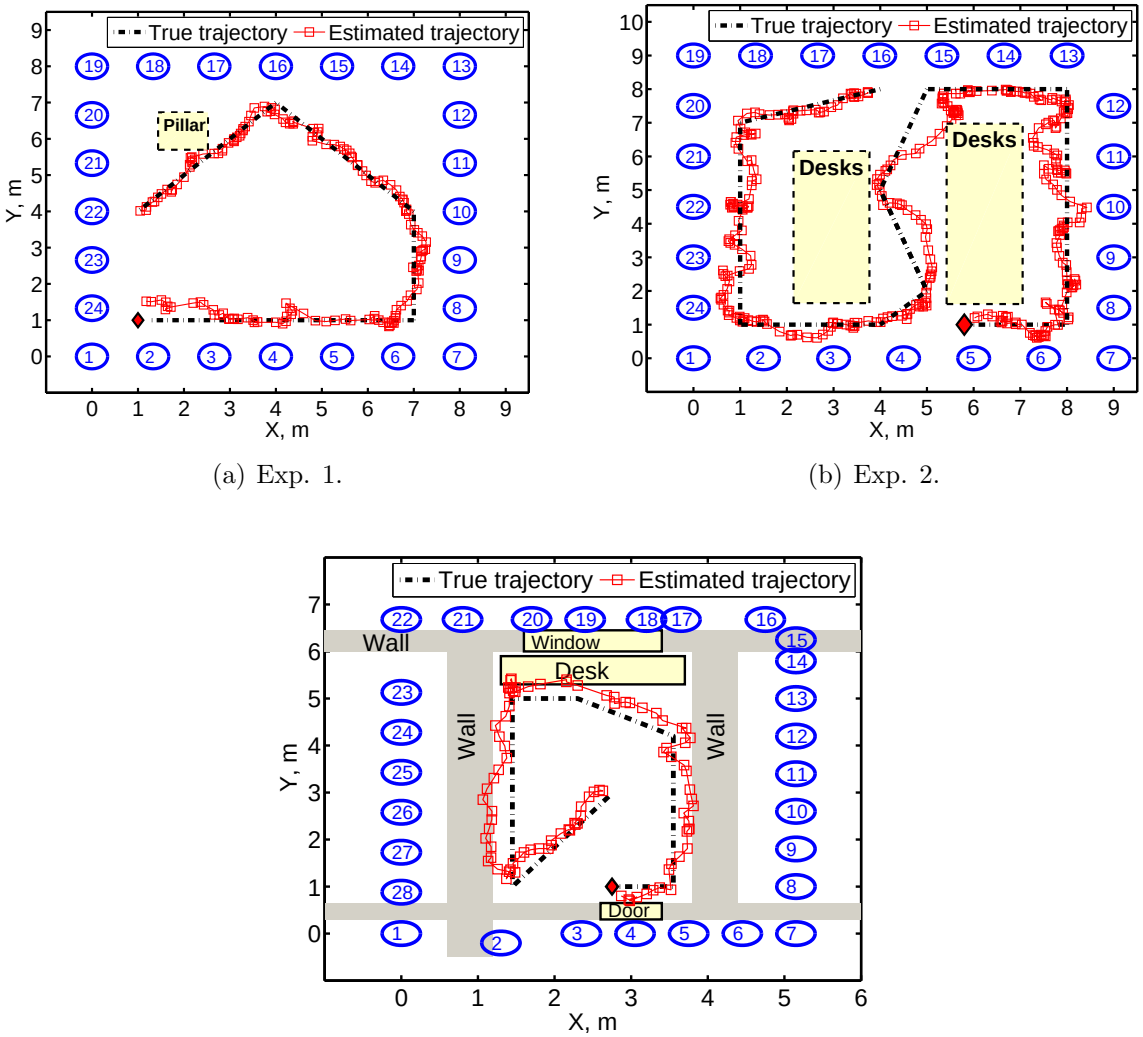
the measurement noise to be Gaussian, we do not perform its skew-Laplacian analysis. The tracking performance in terms of average OSPA error for the testing data set is summarized in Table 3.6.

	SIR			MCMC			PHD
	Average error(m), σ_v = 0.1						
Experiment	Mag	SL	SL [60]	Mag	SL	SL [60]	Mag
Exp. 1	**0.31**	0.35	2.73	**0.31**	0.40	1.06	0.45
Exp. 2	**0.41**	1.01	3.10	**0.37**	0.46	1.50	0.50
Exp. 3	**0.36**	0.56	1.55	**0.30**	0.37	0.37	0.46
	Average error(m), σ_v = 0.2						
Exp. 1	**0.35**	0.38	0.54	**0.32**	0.40	0.34	0.53
Exp. 2	**0.43**	0.47	0.59	**0.41**	0.45	0.46	0.56
Exp. 3	**0.34**	0.45	0.41	0.30	0.36	**0.29**	0.54

Table 3.6 Single target: Average OSPA error (in meter) using different tracking algorithms and different measurement models for Exp. 1, Exp. 2, and Exp. 3. $\sigma_v = 0.1, 0.2$. Error is reported for the testing data set.

Two different values of the innovation noise standard deviation, $\sigma_v = 0.1$ and $\sigma_v = 0.2$, are considered to analyze the robustness of the measurement models. A higher value of σ_v implies less confidence in the motion model; it also facilitates recovery from tracking errors. Table 3.6 suggests the original skew-Laplace model performance is sensitive to the choice of σ_v. In most cases the magnitude model has a lower error than the modified skew-Laplace model, which in turn generally performs better than the original skew-Laplace model. The magnitude model is relatively insensitive to the choice of σ_v for both the SIR and MCMC algorithms. The SIR and MCMC have similar performance and both perform better than the PHD filter. Figures 3.18(a), 3.18(b) and 3.18(c) plot sample target trajectories obtained using the SIR filter and the magnitude model for the three different experiments. For the most part the estimated target trajectories closely follow the true target trajectories with accuracy dropping in presence of obstacles such as the region between the desks in Figure 3.18(b).

Figure 3.19 shows the OSPA error over time using a box-and-whisker diagram for the MCMC algorithm ($\sigma_v = 0.1$). Boxes range from the 25^{th} to 75^{th} percentile, the horizontal bar within the box indicates the median value, and the red pluses indicate outliers. Using the original skew-Laplace model there are multiple lost tracks leading to frequent outliers. For



(a) Exp. 1.

(b) Exp. 2.

(c) Exp. 3.

Fig. 3.18 Sample single target trajectories using the magnitude model in different indoor environments. The small diamond indicates the start of trajectory.

the modified skew-Laplace model the performance is comparable to the magnitude model. The computational time requirements of the tracking algorithms also play an important role in their practical applicability. Table 3.7 summarizes the average normalized processing times of each of the algorithms with different measurement model combinations. The normalized processing time is the ratio of time required to process the data to the duration

of the experiment. A normalized time less than 1 indicates a real-time performance of the algorithm. The SIR and PHD algorithms are computationally fast and can perform real time tracking when a single target is present.

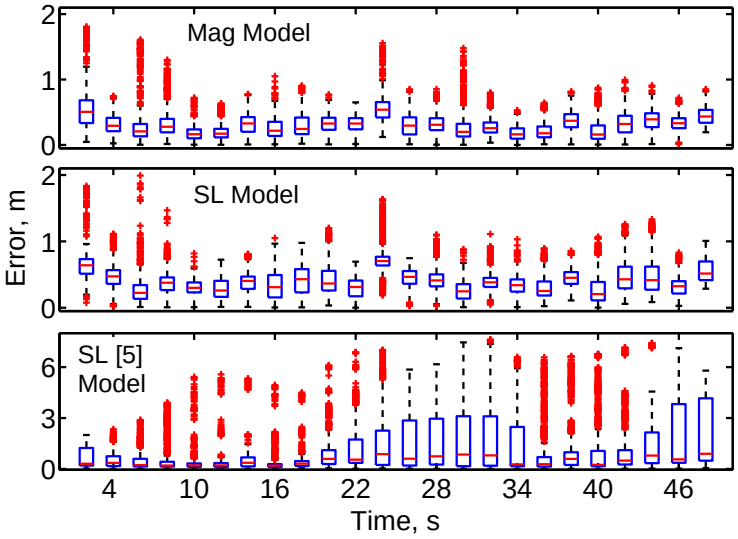


Fig. 3.19 Box-and-whisker plot of the OSPA error over time for the Exp.1 data using the MCMC algorithm.

	SIR			MCMC			PHD
Exp.	Mag	SL	SL [60]	Mag	SL	SL [60]	Mag
Exp. 1	**0.19**	0.45	0.50	6.79	22.79	26.06	0.38
Exp. 2	**0.11**	0.24	0.37	13.19	22.70	25.11	0.63
Exp. 3	**0.12**	0.34	0.41	7.79	13.58	15.80	0.46

Table 3.7 Single target: Average normalized processing time using different tracking algorithms and different measurement models for Exp. 1, Exp. 2, and Exp. 3.

Multiple target tracking with known and fixed number of targets

We now consider tracking experiments in which there are multiple (two or more) targets and the number of targets is fixed and known. Two-target experiments were performed at Setup1 and Setup2 and a three-target experiment was conducted at Setup1. We examined the performance of the skew-Laplace measurement model for multiple targets but it

frequently leads to lost tracks and has significantly higher average error for all algorithms. Hence, in this section, we only report results for the multi-target magnitude measurement model discussed in Section 3.2.3. We use the model parameters $\sigma_w = 2$, $\phi = 3$ and $\sigma_\lambda = 0.4$.

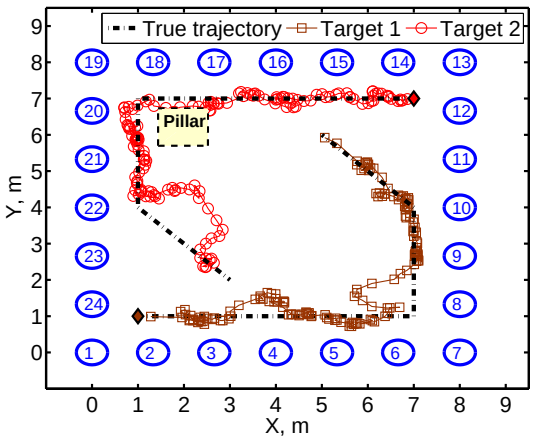
Table 3.8 reports the average OSPA errors for the two-target experiments, Exp. 4 and Exp. 5. To examine the effect of prior information on the overall tracking performance, we consider two different initial particle distributions. For the "Informed Prior", we initialize the particles at the first time step according to a Gaussian distribution with variance 1, centered at the true target locations. In the "Uniform Prior", we initialize the particles uniformly at random within the observation region.

		Informed Prior				Uniform Prior			
Experiment	N_{ppt}	SIR	MPF	MCMC	PHD	SIR	MPF	MCMC	PHD
Exp. 4	100	1.46	1.00	**0.86**	0.93	1.63	1.78	1.06	**0.96**
	250	1.05	**0.79**	0.83	0.86	1.27	1.58	1.00	**0.89**
	500	0.92	**0.67**	0.80	0.88	1.06	1.44	0.97	**0.91**
	750	0.80	**0.62**	0.80	0.88	1.01	1.36	0.92	**0.91**
Exp. 5	100	1.11	0.94	**0.72**	0.93	1.15	1.21	**0.74**	0.92
	250	1.00	0.82	**0.70**	0.92	0.94	1.01	**0.72**	0.89
	500	0.83	**0.71**	**0.70**	0.89	0.88	0.92	**0.72**	0.91
	750	0.72	**0.70**	**0.69**	0.88	0.80	1.00	**0.72**	0.89

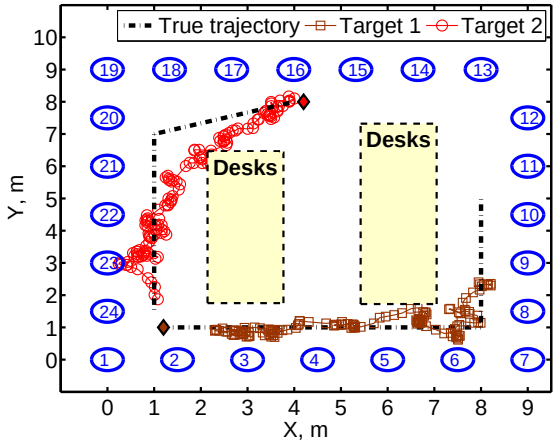
Table 3.8 Two targets: Average OSPA error (in meters) using different algorithms with $N_{ppt} = 100, 250, 500$ & 750 for Exp. 4 and Exp. 5.

The MPF performs best when the "Informed Prior" is used. For all the methods except the PHD the average error reduces as the number of particles is increased. There is minimal gain in accuracy for any algorithm if the number of particles per target is increased beyond $N_{ppt} = 750$. The error fluctuations for the PHD filter with respect to the number of particles are due primarily to errors during the clustering stage. When the particles are initialized using the non-informative "Uniform Prior", the SIR and MPF algorithm performances worsen significantly for Experiment 4. The MCMC algorithm is more robust with respect to the prior information, and the PHD filter displays little sensitivity to the initialization.

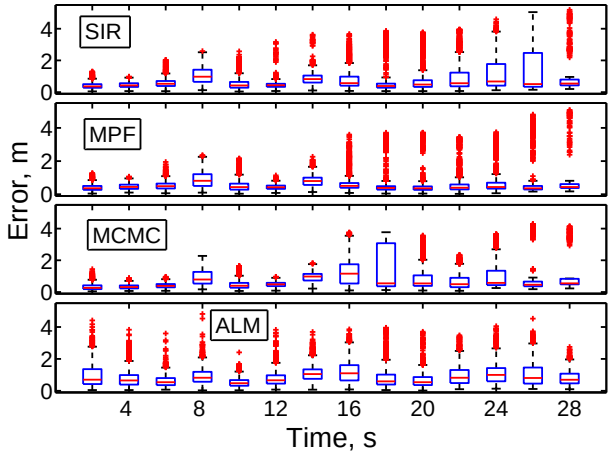
Figures 3.20(a) and 3.20(b) show sample target trajectories obtained using the MPF algorithm in Exp. 4 and Exp. 5 respectively. Figure 3.20(c) displays box-and-whisker OSPA error plots for the Exp. 4 data for the four tracking algorithms. The median error is lowest for the MPF algorithm. The computational requirements (normalized processing



(a) Exp. 4, Target tracks



(b) Exp. 5, Target tracks



(c) Exp. 4, Box-and-whisker plot

Fig. 3.20 (a), (b): Example target tracks estimated by the MPF algorithm when two targets are present for Exp. 4 and Exp. 5. (c): Box-and-whisker plot of OSPA error over time for Exp. 4 data.

time) of the different algorithms are displayed in Table 3.9. Results for both cases "Informed Prior" and "Uniform Prior" were observed to be similar and hence we report only the latter. The SIR, MPF (500 particles or less) and PHD filter (250 particles or less) can execute in real-time. The MCMC algorithm has a major computational overhead and cannot execute in real-time with the current implementation and processor.

Table 3.10 reports the average OSPA error for Exp. 6 data when three targets are

Experiment	N_{ppt}	SIR	MPF	MCMC	PHD
Exp. 4	100	**0.12**	**0.11**	16.42	**0.61**
	250	**0.34**	**0.30**	25.97	**0.95**
	500	**0.75**	**0.68**	41.81	1.57
	750	1.23	1.10	57.41	2.21
Exp. 5	100	**0.12**	**0.11**	16.37	**0.56**
	250	**0.34**	**0.30**	25.90	**0.88**
	500	**0.76**	**0.67**	41.78	1.44
	750	1.21	1.10	57.68	2.02

Table 3.9 Two targets: Average normalized processing time using different algorithms with N_{ppt} = 100, 250, 500 & 750 for Exp. 4 and Exp. 5 (using uniform prior).

simultaneously present. When using the "Informed Prior", the MPF algorithm has the lowest tracking error. When the "Uniform Prior" is used, the PHD filter performs noticeably better than the other algorithms. The PHD filter operates in the single-target state space and hence increasing the number of targets has less impact on its performance. We observe that there is a slight increase in error of the PHD filter as the number of particles is increased. A similar trend was observed even after increasing the number of Monte Carlo trials. A possible explanation for this behaviour could be that the additional particles disproportionately populate the state space region corresponding to only some of the targets. This can in turn introduce a bias in the mean vector and the covariance matrix which we estimate using particles (see Figure 3.12).

Exp. 6	Informed Prior				Uniform Prior			
N_{ppt}	SIR	MPF	MCMC	PHD	SIR	MPF	MCMC	PHD
100	1.94	**0.75**	0.90	0.75	2.08	1.20	0.98	**0.76**
250	1.34	**0.58**	0.89	0.76	1.56	1.07	0.92	**0.77**
500	1.04	**0.56**	0.84	0.80	1.09	0.96	0.97	**0.83**
750	0.91	**0.54**	0.83	0.83	0.95	1.03	0.91	**0.84**

Table 3.10 Three targets: Average OSPA error (in meters) for Exp. 6, Setup1.

A sample target trajectory obtained using the MPF algorithm is shown in Figure 3.21. The corresponding computational time requirements (normalized processing time) for the algorithms are summarized in Table 3.11. Real-time tracking is possible with the SIR and

MPF algorithms using 250 particles or less, but this involves some decrease in accuracy. The PHD filter can execute in real time for 100 particles, but its accuracy is less sensitive to the number of particles.

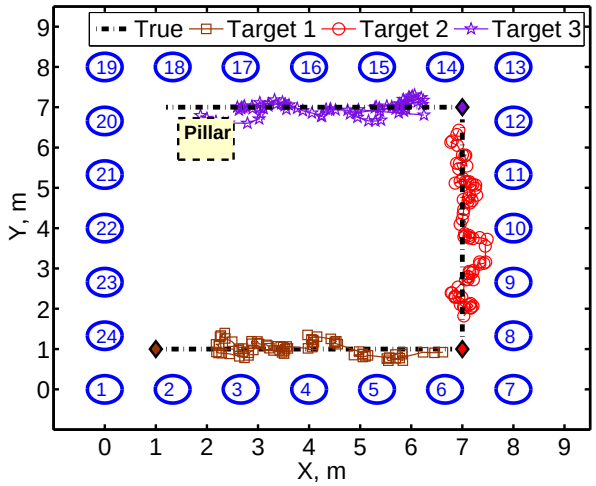


Fig. 3.21 Three targets: Example target tracks estimated by the MPF algorithm when three targets are present for Exp. 6, Setup1.

N_{ppt}	SIR	MPF	MCMC	PHD
100	**0.29**	**0.25**	31.49	**0.86**
250	**0.83**	**0.69**	54.51	1.48
500	1.94	1.58	93.63	2.59
750	2.95	2.45	131.4	3.62

Table 3.11 Three targets: Average normalized processing time for Exp. 6, Setup1 (using uniform prior).

Multiple target tracking with unknown and varying number of targets

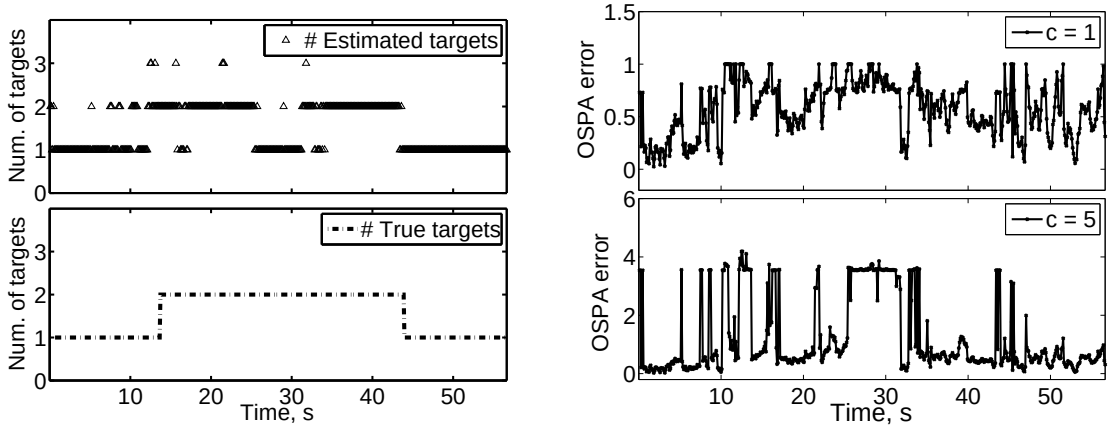
We now address the most challenging indoor target tracking task of tracking a time varying and unknown number of targets. We use the SIR and MCMC algorithms, adapted to account for varying target number as discussed in Section 3.3, to track the targets. The multi-target magnitude measurement model is used for likelihood computation. We assume the maximum number of targets is $N_{max} = 4$. The model parameters are $\sigma_w = 2$, $\phi = 5$ and $\sigma_v = 0.1$.

Table 3.12 shows the average OSPA error values for the Exp. 7 data for different values of the cardinality penalty c. We set the number of particles per target to $N_{ppt} = 500$ and use the "Informed Prior" particle initialization. When the cardinality penalty is small ($c = 1$), the error is comparable to the two-target error when the number of targets is known, indicating that the tracking performance of the algorithms is not significantly affected by the missing information. Both the algorithms have similar error values for small c and MCMC error increases slightly with c indicating the MCMC cardinality estimate is not as robust as the SIR.

Exp. 7	Average OSPA error		
Algorithm	$c = 1$	$c = 2.5$	$c = 5$
SIR	0.60	0.91	1.32
MCMC	0.60	0.93	1.38

Table 3.12 Varying target number: Average OSPA errors for Exp. 7, Setup2.

Figure 3.22 compares the actual number of targets to the SIR cardinality estimate. Also shown is the corresponding error variation for $c = 1$ and $c = 5$. The algorithm makes numerous cardinality estimation errors, particularly in the time-period 25-30s, when one of the targets is in the region between the desks (Figure 3.16(c)) and hence is more difficult to detect.



(a) Target number estimate using SIR

(b) OSPA error with $c = 1$ and $c = 5$

Fig. 3.22 (a) Example of the true number of targets and the SIR cardinality estimate (Exp. 7, Setup2). (b) Average OSPA error for $c = 1$ and $c = 5$.

3.6 Conclusions

In this chapter we described the radio frequency tomography approach for multitarget tracking. We have presented measurement models for both outdoor and indoor environments. Data collected from multiple outdoor and indoor locations was used to experimentally validate the additive attenuation of multitarget measurement models. The experiments indicated that the models adequately capture the average attenuation; the noise, although close to zero-mean, is not well described by the Gaussian or truncated Gaussian model as the observed heavy tails are not captured. This does not substantially compromise performance in the reported experiments because the sensors are relatively densely deployed.

We compared four particle based multitarget tracking algorithms. Using data from experiments at multiple sites representing a variety of measurement challenges, we demonstrated that the algorithms can successfully track up to four targets in outdoor environment and up to three targets in indoor environment. The system and algorithms can perform well when sensors are deployed in uncluttered outdoor locations, inside a room with furniture, or densely deployed outside a small, uncluttered room with thick concrete walls. The algorithms can track a time-varying number of targets, but struggle to estimate the number of targets accurately.

Chapter 4

Moment filters for superpositional sensors

In the multitarget tracking problem often the number of targets and the number of observations are unknown and time varying. Representing the multitarget target state and observations as vectors is inefficient in such scenarios. Tracking algorithms have to search through a much larger space spanning multiple dimensions which make them slow and inaccurate. Assuming that the order of elements is not a significant concern, a more suitable representation in such a case is the finite set. This is the basis for the *random finite set* (RFS) framework [9, 33] which represents target states and observations as realizations of random finite sets.

Moment based filters developed using the random finite set theory propagate the first and/or higher order moments of the multitarget density function over time. Examples include the *probability hypothesis density* (PHD) filter and the *cardinalized probability hypothesis density* (CPHD) filter. We study these filters in the context of superpositional sensor measurement model discussed in Section 2.4.2. The general CPHD filter for superpositional sensors was first derived by Mahler [14] but the filter equations are computationally intractable. The computationally tractable PHD update equations were first derived by Thouin, Nannuru and Coates in [4, 153]; the computationally tractable CPHD update equations were first derived by Mahler and El-Fallah in [38]. For completeness, in this chapter we first present the derivations of computationally tractable approximations of the PHD [4, 153] and CPHD [38] filters for superpositional sensors. As a novel contribution of

this thesis, later in this chapter we develop auxiliary particle filter based implementations of the PHD and CPHD filters and perform numerical simulations.

This chapter is organized as follows. In Section 4.1 we introduce the moment filters and outline the derivation methodology. In Section 4.2, we derive computationally tractable approximate PHD filter update equations for superpositional sensors. Approximate update equations for the CPHD filter are derived in Section 4.3. Section 4.4 discusses the auxiliary particle filter based implementations of the PHD and CPHD filters. We conduct performance analysis of the PHD, CPHD and MCMC filters using numerical simulations in Section 4.5. Section 4.6 provides conclusions for this chapter.

4.1 Moment filters

The optimal Bayes filter solution (Section 2.5) for the multitarget tracking problem is analytically intractable because the related set integrals cannot be evaluated except for very few simple cases. One method to solve this problem is to make certain simplifying assumptions or approximations about the involved multitarget distributions. Under these assumptions it suffices to propagate the relevant sufficient statistics over time reducing the complexity of the filter equations. This is the main idea behind the development of the PHD and CPHD filters.

4.1.1 PHD filter

The PHD filter propagates the probability hypothesis density function (Section 2.2.3) over time. When the multitarget distributions are Poisson (Section 2.3.1) the PHD filter is the optimal filter since a Poisson random finite set can be completely characterized by its PHD function. The PHD filter was first developed for standard sensors by Mahler [86]. The general PHD filter equations for superpositional sensors were first derived by Mahler [14] (the general CPHD filter equations are explicitly provided; PHD filter equations can be derived as a special case). These equations are computationally intractable and are of very limited practical importance. To overcome this limitation, approximate but computationally tractable PHD filter equations for superpositional sensors were derived by Thouin, Nannuru and Coates in [4, 153].

4.1.2 CPHD filter

In the PHD filter, since only the probability hypothesis density function is propagated over time, there is an inherent assumption that the cardinality distribution of the multitarget set is Poisson. This is not a desirable assumption especially since the variance of the Poisson distribution is equal to its mean implying that as the number of targets increases, the error in its estimation becomes larger. To overcome this problem the cardinalized probability hypothesis density filter was derived by Mahler [87] for standard sensors. The CPHD filter assumes that the multitarget state can be modeled by an IIDC random finite set (Section 2.3.2) and hence along with the the probability hypothesis density function it also propagates the cardinality distribution over time. This provides more accurate cardinality information about the multitarget state. The general CPHD filter equations for superpositional sensors were first derived by Mahler in [14] and their computationally tractable approximate versions were derived by Mahler and El-Fallah in [38].

4.1.3 Key steps of derivation

The primary steps in the derivations of computationally tractable approximations of the PHD and CPHD filter update equations for superpositional sensors are the application of

a) Change of variables formula (Section 2.2.4);

b) Campbell's theorem (Section 2.2.5); and

c) Gaussian approximation for intermediate distributions.

The PHD and CPHD filter update equations for superpositional sensors can be expressed as set integrals, but these are computationally intractable. Application of suitable change of variables formulae allows us to transform these set integrals into ordinary integrals.

Although much simpler, evaluation of the resultant ordinary integrals remains an unpalatable computational challenge. Application of Campbell's theorem (the linear case for the PHD and the more general quadratic version for the CPHD) allows us to evaluate the mean and variance of the intermediate distributions. We can then use a Gaussian approximation of this distribution. When combined with Gaussian sensor noise assumption this leads to approximate update equations that involve much less computational overhead.

We note that the derivation approach we present of the approximate PHD and CPHD filters in this chapter and the approximate multi-Bernoulli filters in Chapter 5 for superpositional sensors is significantly different from that of the same filters for standard sensors [34]. The latter uses the probability generating functional based formulation of the multitarget Bayes filter (Section 2.5), which cannot be easily applied for the superpositional observation model. The approximate approach developed in these chapters cannot be extended to the standard observation model because the crucial step of applying Campbell's theorem can only be used when the observation model has a superpositional form.

4.2 PHD filter

Let $\mathcal{X}$ be the single target state space and let $\mathcal{Y}$ be the space of observations. If Z_k is the observation vector and $X_k = \{\mathbf{x}_{1,k}, \mathbf{x}_{2,k}, \ldots, \mathbf{x}_{n_k,k}\}$ is the multitarget state at time k then we have $\mathbf{x}_{i,k} \in \mathcal{X}$ and $Z_k \in \mathcal{Y}$. We make the following modelling assumptions while deriving the superpositional sensor PHD filter equations:

a) Target birth at time $k+1$ is modelled using a Poisson random finite set.

b) The predicted multitarget distribution at time $k+1$ is Poisson.

c) The sensor observation model is of the superpositional form as described in Section 2.4.2.

Let $D_{k+1|k}(\mathbf{x}|Z_{1:k})$ and $D_{k+1|k+1}(\mathbf{x}|Z_{1:k+1})$ be the predicted and posterior PHD functions at time $k+1$. They are related to the predicted and posterior multitarget densities as follows

$$D_{k+1|k}(\mathbf{x}|Z_{1:k}) = \int_{\mathcal{X}} f_{k+1|k}(\{\mathbf{x}\} \cup W|Z_{1:k})\, \delta W \tag{4.1}$$

$$D_{k+1|k+1}(\mathbf{x}|Z_{1:k+1}) = \int_{\mathcal{X}} f_{k+1|k+1}(\{\mathbf{x}\} \cup W|Z_{1:k+1})\, \delta W \tag{4.2}$$

For brevity, in the following subsections, we drop the explicit notation of conditional dependence on the observations. For example, we write:

$$D_{k+1|k}(\mathbf{x}) \equiv D_{k+1|k}(\mathbf{x}|Z_{1:k}), \tag{4.3}$$

$$D_{k+1|k+1}(\mathbf{x}) \equiv D_{k+1|k+1}(\mathbf{x}|Z_{1:k+1}) \tag{4.4}$$

We also define the normalized predictive PHD $s_{k+1|k}(\mathbf{x})$ as:

$$s_{k+1|k}(\mathbf{x}) \stackrel{\text{def}}{=} \frac{D_{k+1|k}(\mathbf{x})}{\mu_{k+1|k}} \tag{4.5}$$

$$\mu_{k+1|k} \stackrel{\text{def}}{=} \int_{\mathcal{X}} D_{k+1|k}(\mathbf{x}) d\mathbf{x}. \tag{4.6}$$

Denote the superpositional sensor multitarget likelihood function as $h_{\mathbf{z}_{k+1}}(X)$. See Section 2.4.2 for a more detailed discussion of this likelihood function.

4.2.1 Prediction step

The superpositional assumption on the likelihood model does not affect the time prediction step of the filter. Hence we can apply Mahler's general law of motion for PHDs to compute the predictive PHD [86],

$$D_{k+1|k}(\mathbf{x}) = b_{k+1|k}(\mathbf{x}) + \int_{\mathcal{X}} p_{sv,k+1}(\mathbf{w}) t_{k+1|k}(\mathbf{x}|\mathbf{w}) D_{k|k}(\mathbf{w}) d\mathbf{w} \tag{4.7}$$

where $p_{sv,k+1}(\mathbf{w})$ is the survival probability of target located at $\mathbf{w}$ at time $k+1$; $b_{k+1|k}(\mathbf{x})$ is the PHD of the target birth process at $\mathbf{x}$; and $t_{k+1|k}(\mathbf{x}|\mathbf{w})$ is the single target Markovian transition function. We do not consider spawning of new targets from existing targets in our analysis.

4.2.2 Update step

We now derive the update formula for the PHD filter. By definition, the posterior PHD at time $k+1$ is given by

$$D_{k+1|k+1}(\mathbf{x}) = \int_{\mathcal{X}} f_{k+1|k+1}(\{\mathbf{x}\} \cup W) \delta W \tag{4.8}$$

Applying Bayes rule we have:

$$D_{k+1|k+1}(\mathbf{x}) = \int_{\mathcal{X}} \frac{h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f_{k+1|k}(\{\mathbf{x}\} \cup W)}{f(\mathbf{z}_{k+1}|Z_{1:k})} \delta W \tag{4.9}$$

$$= \frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f_{k+1|k}(\{\mathbf{x}\} \cup W) \, \delta W}{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(W) f_{k+1|k}(W) \delta W}. \tag{4.10}$$

If we now assume that the predicted multitarget distribution $f_{k+1|k}(W)$ has a Poisson distribution with the PHD function $D_{k+1|k}(\mathbf{x})$, we can write

$$f_{k+1|k}(\{\mathbf{x}\} \cup W) = e^{-\mu_{k+1|k}} D_{k+1|k}(\mathbf{x}) \prod_{\mathbf{w} \in W} D_{k+1|k}(\mathbf{w}) \tag{4.11}$$

$$= D_{k+1|k}(\mathbf{x})\, f_{k+1|k}(W) \tag{4.12}$$

Thus from (4.10)

$$D_{k+1|k+1}(\mathbf{x}) = D_{k+1|k}(\mathbf{x}) \frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f_{k+1|k}(W) \delta W}{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(W) f_{k+1|k}(W) \delta W} \tag{4.13}$$

$$= D_{k+1|k}(\mathbf{x})\, L_{\mathbf{z}_{k+1}}(\mathbf{x}) \tag{4.14}$$

where we have defined the pseudolikelihood function $L_{\mathbf{z}_{k+1}}(\mathbf{x})$ as

$$L_{\mathbf{z}_{k+1}}(\mathbf{x}) \stackrel{\text{def}}{=} \frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f_{k+1|k}(W)\, \delta W}{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(W) f_{k+1|k}(W)\, \delta W} \tag{4.15}$$

Under the assumption of Gaussian sensor noise and the superpositional sensor model from Section 2.4.2,

$$h_{\mathbf{z}_{k+1}}(W) = \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - \zeta(W)) \tag{4.16}$$

$$h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) = \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \zeta(W)) \tag{4.17}$$

$$L_{\mathbf{z}_{k+1}}(\mathbf{x}) = \frac{\int_{\mathcal{X}} \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \zeta(W)) f_{k+1|k}(W)\, \delta W}{\int_{\mathcal{X}} \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - \zeta(W)) f_{k+1|k}(W)\, \delta W} \tag{4.18}$$

Applying the change of variable $\mathbf{y} = \zeta(W)$ in the numerator and denominator and using the formula in equation (2.17) for change of variables from Section 2.2.4 leads to:

$$L_{\mathbf{z}_{k+1}}(\mathbf{x}) = \frac{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \mathbf{y})\, Q_{k+1|k}(\mathbf{y})\, d\mathbf{y}}{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - \mathbf{y})\, Q_{k+1|k}(\mathbf{y})\, d\mathbf{y}} \tag{4.19}$$

To develop a computationally tractable update equation we approximate $Q_{k+1|k}(\mathbf{y})$ using a Gaussian distribution, $Q_{k+1|k}(\mathbf{y}) \approx \mathcal{N}_{\Sigma_{k+1}}(\mathbf{y} - \mathbf{m}_{k+1})$. The parameters $\mathbf{m}_{k+1}$ and Σ_{k+1} can be obtained using the Campbell's theorem from Section 2.2.5. For the Poisson process we have $D_{k+1|k}(\mathbf{x}) = \mu_{k+1|k}\, s_{k+1|k}(\mathbf{x})$ and $D_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\}) = D_{k+1|k}(\mathbf{x}_1) D_{k+1|k}(\mathbf{x}_2)$. Applying

Campbell's theorem gives

$$\mathbf{m}_{k+1} = \int_{\mathcal{X}} g(\mathbf{x})\, D_{k+1|k}(\mathbf{x}) d\mathbf{x} \tag{4.20}$$

$$= \mu_{k+1|k} \int_{\mathcal{X}} g(\mathbf{x})\, s_{k+1|k}(\mathbf{x}) d\mathbf{x} \tag{4.21}$$

$$= \mu_{k+1|k}\, \hat{\mathbf{m}}_{k+1} \tag{4.22}$$

$$\Sigma_{k+1} = \int_{\mathcal{X}} g(\mathbf{x})\, g(\mathbf{x})^T D_{k+1|k}(\mathbf{x}) d\mathbf{x} \tag{4.23}$$

$$= \mu_{k+1|k} \int_{\mathcal{X}} g(\mathbf{x})\, g(\mathbf{x})^T s_{k+1|k}(\mathbf{x}) d\mathbf{x} \tag{4.24}$$

$$= \mu_{k+1|k}\, \hat{\Sigma}_{k+1} \tag{4.25}$$

where

$$\hat{\mathbf{m}}_{k+1} \stackrel{\text{def}}{=} \int_{\mathcal{X}} g(\mathbf{x})\, s_{k+1|k}(\mathbf{x}) \tag{4.26}$$

$$\hat{\Sigma}_{k+1} \stackrel{\text{def}}{=} \int_{\mathcal{X}} g(\mathbf{x}) g(\mathbf{x})^T\, s_{k+1|k}(\mathbf{x}) \tag{4.27}$$

The pseudo-likelihood $L_{\mathbf{z}_{k+1}}(\mathbf{x})$ can then be approximated as:

$$L_{\mathbf{z}_{k+1}}(\mathbf{x}) \approx \frac{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \mathbf{y})\, \mathcal{N}_{\Sigma_{k+1}}(\mathbf{y} - \mathbf{m}_{k+1}) d\mathbf{y}}{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - \mathbf{y})\, \mathcal{N}_{\Sigma_{k+1}}(\mathbf{y} - \mathbf{m}_{k+1}) d\mathbf{y}} \tag{4.28}$$

$$= \frac{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma_{k+1}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \mathbf{m}_{k+1})}{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma_{k+1}}(\mathbf{z}_{k+1} - \mathbf{m}_{k+1})} \tag{4.29}$$

The approximate update equation for the PHD filter is then

$$D_{k+1|k+1}(\mathbf{x}) \approx L_{\mathbf{z}_{k+1}}(\mathbf{x})\, D_{k+1|k}(\mathbf{x}). \tag{4.30}$$

Thus the update equation involves calculating the ratio of two Gaussian densities. The main computational challenge is evaluating the integrals in (4.26) and (4.27) which require numerical approximation.

Analysis of Gaussian approximation

The approximation $Q_{k+1|k}(\mathbf{y}) \approx \mathcal{N}_{\Sigma_{k+1}}(\mathbf{y} - \mathbf{m}_{k+1})$ has been introduced in order to analytically evaluate the integrals in (4.19). Without this approximation the update equation would

involve integrals which have to be numerically evaluated and it makes the filter implementation computationally demanding. A brief numerical analysis of the errors introduced in the calculation of the integral and the pseudo-likelihood because of this approximation is provided in Appendix A. To summarize, we observe that as the average number of targets represented by the underlying random finite sets is increased the error between the original and the approximated integral reduces significantly.

4.3 CPHD filter

The CPHD filter propagates both the probability hypothesis density and the cardinality distribution over time. We make the following modelling assumptions while deriving the superpositional sensor CPHD filter equations:

a) Target birth at time $k+1$ is modelled using an IIDC random finite set.

b) The predicted multitarget distribution at time $k+1$ is IIDC.

c) The sensor observation model is of the superpositional form as described in Section 2.4.2.

Let $D_{k+1|k}(\mathbf{x})$ and $D_{k+1|k+1}(\mathbf{x})$ be the predicted and the posterior PHD functions at time $k+1$. Let the predicted and posterior cardinality distributions at time $k+1$ be $\pi_{k+1|k}(n) \equiv \pi_{k+1|k}(n|Z^{[k]})$ and $\pi_{k+1|k+1}(n) \equiv \pi_{k+1|k+1}(n|Z^{[k+1]})$ respectively. Let $s_{k+1|k}(\mathbf{x})$ be the normalized predicted PHD and $\mu_{k+1|k}$ be the mean predicted cardinality

$$s_{k+1|k}(\mathbf{x}) \stackrel{\text{def}}{=} \frac{D_{k+1|k}(\mathbf{x})}{\mu_{k+1|k}} \tag{4.31}$$

$$\mu_{k+1|k} \stackrel{\text{def}}{=} \sum_{n=1}^{\infty} n\, \pi_{k+1|k}(n) = \int_{\mathcal{X}} D_{k+1|k}(\mathbf{x}) d\mathbf{x} \tag{4.32}$$

Denote the superpositional sensor multitarget likelihood function as $h_{\mathbf{z}_{k+1}}(X)$.

4.3.1 Prediction step

The PHD and cardinality prediction equations for superpositional sensors are the same as those for standard sensors since the likelihood function has no role. The PHD function prediction equation for the CPHD filter is the same as that for the PHD filter which we

repeat here for completeness

$$D_{k+1|k}(\mathbf{x}) = b_{k+1|k}(\mathbf{x}) + \int_{\mathcal{X}} p_{sv,k+1}(\mathbf{w}) t_{k+1|k}(\mathbf{x}|\mathbf{w}) D_{k|k}(\mathbf{w}) d\mathbf{w} \tag{4.33}$$

where $p_{sv,k+1}(\mathbf{w})$ is the survival probability of target located at $\mathbf{w}$ at time $k+1$; $b_{k+1|k}(\mathbf{x})$ is the PHD of the target birth process at $\mathbf{x}$; and $t_{k+1|k}(\mathbf{x}|\mathbf{w})$ is the single target Markovian transition function.

The cardinality distribution prediction equation is given by [87]:

$$\pi_{k+1|k}(n) = \sum_{j=0}^{n} p_{b,k+1}(n-j) \sum_{l=j}^{\infty} \binom{l}{j} \frac{\langle D_{k+1|k}, p_{sv,k+1} \rangle^j \langle D_{k+1|k}, 1 - p_{sv,k+1} \rangle^{l-j}}{\langle D_{k+1|k}, 1 \rangle^l} \pi_{k|k}(l) \tag{4.34}$$

where $p_{b,k+1}(j)$ is the birth probability of j new targets at time $k+1$. When the target survival probability is constant at all times and at all locations, $p_{sv,k}(\mathbf{x}) = p_s$, the above equation reduces to

$$\pi_{k+1|k}(n) = \sum_{j=0}^{n} p_{b,k+1}(n-j) \sum_{l=j}^{\infty} \binom{l}{j} p_s^j (1-p_s)^{l-j} \pi_{k|k}(l) \tag{4.35}$$

For the derivations we assume that there exists some $n_0 \geq 0$ such that for all $n > n_0$, we have $\pi_{k+1|k}(n) < 1/n$. This assumption holds in the common case when there is a bound on the maximum number of targets.

4.3.2 Update step

The CPHD filter propagates a probability distribution of the cardinality of the random set representing the state of the system along with the PHD. The derivation of the approximate CPHD filter update equations proceeds along similar lines as that of the PHD filter.

PHD update

As before from (4.10), the posterior PHD at time $k+1$ is given by the expression:

$$D_{k+1|k+1}(\mathbf{x}) = \frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f_{k+1|k}(\{\mathbf{x}\} \cup W) \, \delta W}{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(W) f_{k+1|k}(W) \delta W} \tag{4.36}$$

Assuming that the predicted multitarget density $f_{k+1|k}(W)$ corresponds to an IIDC random finite set we have

$$f_{k+1|k}(\{\mathbf{x}\} \cup W) = (|W|+1)!\, \pi_{k+1|k}(|W|+1)\, s_{k+1|k}(\mathbf{x}) \prod_{\mathbf{w} \in W} s_{k+1|k}(\mathbf{w}) \tag{4.37}$$

$$= \mu_{k+1|k}\, s_{k+1|k}(\mathbf{x})\, |W|! \frac{(|W|+1)\, \pi_{k+1|k}(|W|+1)}{\mu_{k+1|k}} \prod_{\mathbf{w} \in W} s_{k+1|k}(\mathbf{w}) \tag{4.38}$$

$$= D_{k+1|k}(\mathbf{x})\, f^{o}_{k+1|k}(W) \tag{4.39}$$

where

$$f^{o}_{k+1|k}(W) \stackrel{\text{def}}{=} |W|!\, \pi^{o}_{k+1|k}(|W|) \prod_{\mathbf{w} \in W} s_{k+1|k}(\mathbf{w}) \tag{4.40}$$

$$\pi^{o}_{k+1|k}(n) \stackrel{\text{def}}{=} \frac{(n+1)\, \pi_{k+1|k}(n+1)}{\mu_{k+1|k}}. \tag{4.41}$$

From the above formulation $f^{o}_{k+1|k}(W)$ is a multitarget density corresponding to an IIDC RFS with density function $s_{k+1|k}(\mathbf{x})$ and cardinality distribution $\pi^{o}_{k+1|k}(n)$ which is a valid cardinality distribution since

$$\sum_{n=0}^{\infty} \pi^{o}_{k+1|k}(n) = \frac{1}{\mu_{k+1|k}} \sum_{n=0}^{\infty} (n+1)\, \pi_{k+1|k}(n+1) \tag{4.42}$$

$$= \frac{1}{\mu_{k+1|k}} \times \mu_{k+1|k} = 1. \tag{4.43}$$

Thus from (4.36)

$$D_{k+1|k+1}(\mathbf{x}) = D_{k+1|k}(\mathbf{x}) \frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f^{o}_{k+1|k}(W) \delta W}{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(W) f_{k+1|k}(W) \delta W} \tag{4.44}$$

$$= D_{k+1|k}(\mathbf{x})\, L^{o}_{\mathbf{z}_{k+1}}(\mathbf{x}) \tag{4.45}$$

where the pseudolikelihood function is now defined as

$$L^{o}_{\mathbf{z}_{k+1}}(\mathbf{x}) \stackrel{\text{def}}{=} \frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f^{o}_{k+1|k}(W)\, \delta W}{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(W) f_{k+1|k}(W)\, \delta W} \tag{4.46}$$

Under the assumption of Gaussian sensor noise and the superpositional sensor model,

$$L^o_{\mathbf{z}_{k+1}}(\mathbf{x}) = \frac{\int_{\mathcal{X}} \mathcal{N}_{\Sigma_\mathbf{z}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \zeta(W)) f^o_{k+1|k}(W)\, \delta W}{\int_{\mathcal{X}} \mathcal{N}_{\Sigma_\mathbf{z}}(\mathbf{z}_{k+1} - \zeta(W)) f_{k+1|k}(W)\, \delta W} \tag{4.47}$$

Performing the change of variables $\mathbf{y}^o = \zeta(W)$ in the numerator and $\mathbf{y} = \zeta(W)$ in the denominator we get

$$L^o_{\mathbf{z}_{k+1}}(\mathbf{x}) = \frac{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_\mathbf{z}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \mathbf{y}^o) Q^o_{k+1|k}(\mathbf{y}^o)\, d\mathbf{y}}{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_\mathbf{z}}(\mathbf{z}_{k+1} - \mathbf{y}) Q_{k+1|k}(\mathbf{y})\, d\mathbf{y}} \tag{4.48}$$

Approximating the density functions $Q^o_{k+1|k}(\mathbf{y}^o)$ and $Q_{k+1|k}(\mathbf{y})$ to be Gaussian, from Campbell's theorem we have,

$$Q^o_{k+1|k}(\mathbf{y}^o) \approx \mathcal{N}_{\Sigma^o_{k+1}}(\mathbf{y} - \mathbf{m}^o_{k+1}) \tag{4.49}$$

$$Q_{k+1|k}(\mathbf{y}) \approx \mathcal{N}_{\Sigma_{k+1}}(\mathbf{y} - \mathbf{m}_{k+1}) \tag{4.50}$$

$$\mathbf{m}^o_{k+1} = \frac{a}{\mu_{k+1|k}}\, \hat{\mathbf{m}}_{k+1} \tag{4.51}$$

$$\Sigma^o_{k+1} = \frac{a}{\mu_{k+1|k}}\, \hat{\Sigma}_{k+1} - \left(\frac{a^2}{\mu^2_{k+1|k}} - \frac{b}{\mu_{k+1|k}} \right) \hat{\mathbf{m}}_{k+1} \hat{\mathbf{m}}^T_{k+1} \tag{4.52}$$

$$\mathbf{m}_{k+1} = \mu_{k+1|k}\, \hat{\mathbf{m}}_{k+1} \tag{4.53}$$

$$\Sigma_{k+1} = \mu_{k+1|k}\, \hat{\Sigma}_{k+1} - (\mu^2_{k+1|k} - a)\, \hat{\mathbf{m}}_{k+1} \hat{\mathbf{m}}^T_{k+1} \tag{4.54}$$

where

$$\hat{\mathbf{m}}_{k+1} \stackrel{\text{def}}{=} \int_{\mathcal{X}} g(\mathbf{x})\, s_{k+1|k}(\mathbf{x}) \tag{4.55}$$

$$\hat{\Sigma}_{k+1} \stackrel{\text{def}}{=} \int_{\mathcal{X}} g(\mathbf{x}) g(\mathbf{x})^T\, s_{k+1|k}(\mathbf{x}), \tag{4.56}$$

$$a \stackrel{\text{def}}{=} \sum_{n=0}^{\infty} n(n-1) \pi_{k+1|k}(n) \tag{4.57}$$

$$b \stackrel{\text{def}}{=} \sum_{n=0}^{\infty} n(n-1)(n-2) \pi_{k+1|k}(n). \tag{4.58}$$

The expressions for the parameters $\mathbf{m}^o_{k+1}$, Σ^o_{k+1}, $\mathbf{m}_{k+1}$ and Σ_{k+1} are derived in Appendix B.

Hence the approximate pseudo-likelihood function can be expressed as

$$L^o_{\mathbf{z}_{k+1}}(\mathbf{x}) \approx \frac{\mathcal{N}_{\Sigma_\mathbf{z}+\Sigma^o_{k+1}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \mathbf{m}^o_{k+1})}{\mathcal{N}_{\Sigma_\mathbf{z}+\Sigma_{k+1}}(\mathbf{z}_{k+1} - \mathbf{m}_{k+1})}. \tag{4.59}$$

Cardinality update

The posterior cardinality distribution is defined as:

$$\pi_{k+1|k+1}(n) = \int_{|W|=n} f_{k+1|k+1}(W)\delta W \tag{4.60}$$

$$= \frac{\int_{|W|=n} h_{\mathbf{z}_{k+1}}(W)\, f_{k+1|k}(W)\delta W}{\int_\mathcal{X} h_{\mathbf{z}_{k+1}}(W) f_{k+1|k}(W)\delta W} \tag{4.61}$$

$$= \pi_{k+1|k}(n)\, \frac{\int_\mathcal{X} h_{\mathbf{z}_{k+1}}(W)\, f^n_{k+1|k}(W)\delta W}{\int_\mathcal{X} h_{\mathbf{z}_{k+1}}(W) f_{k+1|k}(W)\delta W} \tag{4.62}$$

where

$$f^n_{k+1|k}(W) \stackrel{\text{def}}{=} \frac{1}{\pi_{k+1|k}(n)} \delta_{|W|,n} f_{k+1|k}(W) \tag{4.63}$$

where $\delta_{|W|,n}$ is the Kronecker delta function. The function $f^n_{k+1|k}(W)$ is a valid multitarget distribution as proved in Appendix B. The denominator of (4.62) can be simplified as before. The numerator of (4.62) can be expressed using the superpositional sensor model, Gaussian noise assumption and change of variables formula as

$$\int_\mathcal{X} h_{\mathbf{z}_{k+1}}(W)\, f^n_{k+1|k}(W)\delta W = \int_\mathcal{Y} \mathcal{N}_{\Sigma_\mathbf{z}}(\mathbf{z}_{k+1} - \mathbf{y})\, Q^n_{k+1|k}(\mathbf{y}) d\mathbf{y} \tag{4.64}$$

We approximate the distribution $Q^n_{k+1|k}(\mathbf{y})$ using a Gaussian density function as

$$Q^n_{k+1|k}(\mathbf{y}) \approx \mathcal{N}_{\Sigma^n_{k+1}}(\mathbf{y} - \mathbf{m}^n_{k+1}). \tag{4.65}$$

The parameters $\mathbf{m}^n_{k+1}$ and Σ^n_{k+1} are obtained by applying the Campbell's theorem as

$$\mathbf{m}^n_{k+1} = n\hat{\mathbf{m}}_{k+1} \tag{4.66}$$

$$\Sigma^n_{k+1} = n\,(\hat{\Sigma}_{k+1} - \hat{\mathbf{m}}_{k+1}\hat{\mathbf{m}}^T_{k+1}). \tag{4.67}$$

The expressions for the parameters $\mathbf{m}_{k+1}^n$ and Σ_{k+1}^n are derived in Appendix B. Thus the approximate update expression for the cardinality distribution is

$$\pi_{k+1|k+1}(n) \approx \pi_{k+1|k}(n) \frac{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - \mathbf{y}) \mathcal{N}_{\Sigma_{k+1}^n}(\mathbf{y} - \mathbf{m}_{k+1}^n) d\mathbf{y}}{N_{\Sigma_{\mathbf{z}}+\Sigma_{k+1}}(\mathbf{z}_{k+1} - \mathbf{m}_{k+1})} \tag{4.68}$$

$$= K_c^{-1} \pi_{k+1|k}(n) \frac{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma_{k+1}^n}(\mathbf{z}_{k+1} - \mathbf{m}_{k+1}^n)}{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma_{k+1}}(\mathbf{z}_{k+1} - \mathbf{m}_{k+1})} \tag{4.69}$$

Here K_c is a normalizing factor, included to ensure that the updated cardinality distribution sums to 1. We have:

$$K_c = \sum_{n \geq 0} \pi_{k+1|k}(n) \frac{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma_{k+1}^n}(\mathbf{z}_{k+1} - \mathbf{m}_{k+1}^n)}{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma_{k+1}}(\mathbf{z}_{k+1} - \mathbf{m}_{k+1})} \tag{4.70}$$

The assumption that there is an $n_0 \geq 0$ such that $\pi_{k+1|k}(n) < 1/n$ for all $n > n_0$ ensures that the sum converges and K_c is finite [38].

4.4 Auxiliary particle filter implementations

Equations (4.30), (4.59) and (4.69) give approximate expressions for the time update of the PHD and cardinality when new observation data become available. Although the equations specify how the update should be performed, there are in general no explicit formulae to express the PHD or cardinality at every time step in known standard forms which enable easy computational processing. Hence we consider the particle based implementation of the filters, propagating over time a weighted particle approximation of the PHD (which can be seen as a scaled density). The basic bootstrap particle filter implementation struggles when new targets arrive. We therefore implement an auxiliary particle filter, which, with its look-ahead property, is able to address new target arrivals.

4.4.1 Particle implementation of PHD filter

At every time step k, the PHD is approximated by a weighted set of particles,

$$D_{k|k} \approx \sum_{i=1}^{N_p} w_k^{(i)} \delta(\mathbf{x} - \mathbf{x}_k^{(i)}) \tag{4.71}$$

The particle PHD filter algorithm is described in Figure 4.1. The algorithm first calculates $\hat{N}_{p,k}$, the number of particles used to track the targets that were identified at the previous timestep. In line 3, this is set to the product of the estimated number of targets from the previous timestep, $\hat{N}_{k-1}$ (with $\hat{N}_0 = 0$), and the number of particles allocated to each target, N_{ppt} (an algorithmic parameter).

The auxiliary particle filter implementation consists of the auxiliary proposal step and the weight update step. In the auxiliary proposal step of the PHD filter, the existing particles are propagated, with survival probability $p_{sv,k}(\mathbf{x})$, according to the dynamics. In addition, J_p new particles are added by drawing from an importance sampling distribution $\gamma_{1,k}(\mathbf{x})$ (this distribution could depend on the measurements $\mathbf{z}_k$, but it is in general difficult to construct a meaningful distribution for superpositional sensors). We assume that we can specify a PHD function $p_{b,k}(\mathbf{x})$ for the spontaneous birth process. For the propagated particles, the predictive weights $w_{k|k-1}$ are set to the weights from the previous timestep multiplied by the survival probability, $p_s(\mathbf{x}_{k-1}^{(i)})w_{k-1}$ (line 6, Figure 4.1). The new particles are assigned the importance weight $\frac{p_{b,k}(\mathbf{x}_k^{(i)})}{J_p\gamma_{1,k}(\mathbf{x}_k^{(i)})}$ (line 9). Using these weighted particles, we approximate the integrals in equations (4.56) and (4.55) (line 11) and estimate $\hat{\Sigma}_k$ and $\hat{\mathbf{m}}_k$. These estimates are used to calculate the auxiliary particle weights (line 13).

The weighted particle set thus obtained is used to construct an alternative sampling distribution $\gamma_{2,k}(\mathbf{x})$ for the particles associated with potential new targets (line 17). In our simulations, we use a $\gamma_{2,k}$ formed by drawing particles with probability $(1-p)$ from a prior proportional to the birth intensity function $p_{b,k}$, and with probability p from a Gaussian mixture distribution, $GM\{w_{k|k-1}^{(i)}, \mathbf{x}_k^{(i)}, \Sigma_v\}_{i=\hat{N}_{p,k}+1}^{i=\hat{N}_{p,k}+J_p}$ formed by placing a weighted zero-mean Gaussian with covariance matrix Σ_v at each particle location. The covariance matrix Σ_v is typically diagonal with entries smaller than the process noise variance. The weights in the mixture are the (normalized) particle weights obtained from the auxiliary step. The parameters $\hat{\Sigma}_k$ and $\hat{\mathbf{m}}_k$ are updated using the weighted particle set from the auxiliary step (line 20). The final weight update is performed using equation (4.30) in line 22.

Since the PHD has the property that its integral over the complete observation space is equal to the expected number of targets, we should have $\sum_{i=1}^{N_k} w_k^{(i)} \approx E(|X_k|)$. Due to the approximations made in order to arrive at a computationally tractable filter, however, the error can be substantial. Hence to normalize the weights appropriately, we need to estimate the number of targets from the particles. We use the Silhouette method [154] to

1: Initialize particles $\{w_0^{(i)}, \mathbf{x}_0^{(i)}\}_{i=1}^{i=J_p}$
2: **for** $k = 1$ to T **do**
3: $\hat{N}_{p,k} = \hat{N}_{k-1} \times N_{ppt}$
4: **Auxiliary proposal step**
5: **for** $i = 1$ to $\hat{N}_{p,k}$ **do**
6: *proposal:* $\mathbf{x}_k^{(i)} \sim t_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}), \quad w_{k|k-1}^{(i)} = p_s(\mathbf{x}_{k-1}^{(i)}) w_{k-1}^{(i)}$
7: **end for**
8: **for** $i = \hat{N}_{p,k} + 1$ to $\hat{N}_{p,k} + J_p$ **do**
9: *proposal:* $\mathbf{x}_k^{(i)} \sim \gamma_{1,k}(\mathbf{x}), \quad w_{k|k-1}^{(i)} = \frac{p_{b,k}(\mathbf{x}_k^{(i)})}{J_p \gamma_{1,k}(\mathbf{x}_k^{(i)})}$
10: **end for**
11: $\hat{N}_{k|k-1} = \sum_j w_{k|k-1}^{(j)}$, $\hat{\mathbf{m}}_k = \frac{1}{\hat{N}_{k|k-1}} \sum_j w_{k|k-1}^{(j)} g(\mathbf{x}_k^{(j)})$, $\hat{\Sigma}_k = \frac{1}{\hat{N}_{k|k-1}} \sum_j w_{k|k-1}^{(j)} g(\mathbf{x}_k^{(j)}) g^T(\mathbf{x}_k^{(j)})$
12: **for** $i = 1$ to $\hat{N}_{p,k} + J_p$ **do**
13: *weight update:* $w_{k|k-1}^{(i)} = w_{k|k-1}^{(i)} \times \frac{\mathcal{N}_{\Sigma_{\mathbf{z}} + \Sigma_k}(\mathbf{z}_k - g(\mathbf{x}_k^{(i)}) - \mathbf{m}_k)}{\mathcal{N}_{\Sigma_{\mathbf{z}} + \Sigma_k}(\mathbf{z}_{k+1} - \mathbf{m}_k)}$
14: **end for**
15: **for** $i = \hat{N}_{p,k} + 1$ to $\hat{N}_{p,k} + J_p$ **do**
16: *proposal:* $\mathbf{x}_k^{(i)} \sim \gamma_{2,k}(\mathbf{x}), \quad w_{k|k-1}^{(i)} = \frac{p_{b,k}(\mathbf{x}_k^{(i)})}{J_p \gamma_{2,k}(\mathbf{x}_k^{(i)})}$
17: **end for**
18: Normalize $\{w_{k|k-1}^{(i)}\}_{i=\hat{N}_{p,k}+1}^{i=\hat{N}_{p,k}+J_p}$
19: **Weight update and state estimation**
20: $\hat{N}_{k|k-1} = \sum_j w_{k|k-1}^{(j)}$, $\hat{\mathbf{m}}_k = \frac{1}{\hat{N}_{k|k-1}} \sum_j w_{k|k-1}^{(j)} g(\mathbf{x}_k^{(j)})$, $\hat{\Sigma}_k = \frac{1}{\hat{N}_{k|k-1}} \sum_j w_{k|k-1}^{(j)} g(\mathbf{x}_k^{(j)}) g^T(\mathbf{x}_k^{(j)})$
21: **for** $i = 1$ to $\hat{N}_{p,k} + J_p$ **do**
22: *weight update:* $w_k^{(i)} = w_{k|k-1}^{(i)} \times \frac{\mathcal{N}_{\Sigma_{\mathbf{z}} + \Sigma_k}(\mathbf{z}_k - g(\mathbf{x}_k^{(i)}) - \mathbf{m}_k)}{\mathcal{N}_{\Sigma_{\mathbf{z}} + \Sigma_k}(\mathbf{z}_{k+1} - \mathbf{m}_k)}$
23: **end for**
24: *target number estimation:* $\hat{N}_k = \text{Silhouette}(\{w_k^{(i)}, \mathbf{x}_k^{(i)}\}_{i=1}^{i=\hat{N}_{p,k}+J_p})$
25: *resample step:* $\{w_k^{(i)}, \mathbf{x}_k^{(i)}\}_{i=1}^{i=\hat{N}_{p,k}+J_p} \rightarrow \{\mathbf{x}_k^{(i)}\}_{i=1}^{i=\hat{N}_k \times N_{ppt}}$
26: *clustering step:* $\{\hat{\mathbf{x}}_{k,n}\}_{n=1}^{\hat{N}_k} = \text{cluster}(\{\mathbf{x}_k^{(i)}\}_{i=1}^{i=\hat{N}_k \times N_{ppt}}, \hat{N}_k)$
27: **end for**

Fig. 4.1 Auxiliary particle filter implementation of approximate PHD filter for superpositional sensors.

obtain the target number estimate (line 24).

Given a set of particles partitioned into clusters, the Silhouette method evaluates the appropriateness of the clusters. For the i^{th} particle it calculates $l_1(i)$, the average distance of i to other particles in its cluster and $l_2(i)$, the average distance of i to its neighboring

clusters. The silhouette of the i^{th} particle $l(i)$ is then given by

$$l(i) = \frac{l_2(i) - l_1(i)}{\max(l_1(i), l_2(i))}. \tag{4.72}$$

The average silhouette of all the particles is used to measure the appropriateness of the given partition. The k-means algorithm is used to cluster the particles into clusters. The number of clusters is varied from 2 to N_0. The partition which has the maximum average silhouette is declared to be the cardinality estimate $\hat{N}_k$. The particles are then resampled (line 25) to update the number of particles and clustering is performed to obtain target location estimates (line 26). Other methods can be used to estimate the number of targets from the particle set such as Bayesian information criteria [155] and the Elbow method [156]. These different methods were evaluated in [157] and the silhouette method performed the best among them. Hence we use the silhouette method in our implementations.

4.4.2 Particle implementation of the CPHD filter

Summarized in Figure 4.2 is the particle implementation of the auxiliary CPHD filter. The cardinality distribution is assumed to have a finite support with $\pi_{k|k}(n) = 0,\ n > N_0$. The implementation is much the same as the PHD filter, but we employ the weight update equations for the CPHD filter and the cardinality distribution is also updated. The maximum a posteriori (MAP) estimate of the cardinality is used as the estimate of the number of targets (line 28, Figure 4.2).

4.4.3 Computational complexity

We obtain theoretical computational complexities for the PHD and CPHD algorithms. The major steps in the algorithm implementation are particle propagation, weight update, cardinality prediction and update, resampling and clustering.

For the PHD filter the dominant costs are the weight update and the identification of the number of targets. The weight update involves estimating an $M \times M$ covariance matrix, which has complexity $\mathcal{O}(N_p M^2)$ (where N_p is the number of propagated particles), and computing its inverse, which has complexity $\mathcal{O}(M^3)$. The Silhouette method, used to estimate the number of targets, performs multiple k-means clusterings with complexity $\mathcal{O}(N_p)$ and calculates silhouettes with complexity $\mathcal{O}(N_p^2)$. If the maximum number of

Initialize particles $\{w_0^{(i)}, \mathbf{x}_0^{(i)}\}_{i=1}^{i=J_p}$
for $k = 1$ to T **do**
  $\hat{N}_{p,k} = \hat{N}_{k-1} \times N_{ppt}$
  card. prediction: $\pi^c_{k|k-1}(n) = \sum_{j=0}^{n} p_b(n-j) \left(\sum_{l=j}^{\infty} \binom{l}{r} p_s^j (1-p_s)^{l-j} \pi^c_{k-1|k-1}(l) \right)$
  Auxiliary proposal step
  for $i = 1$ to $\hat{N}_{p,k}$ **do**
    proposal: $\mathbf{x}_k^{(i)} \sim t_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}), \quad w_{k|k-1}^{(i)} = p_s(\mathbf{x}_{k-1}^{(i)}) w_{k-1}^{(i)}$
  end for
  for $i = \hat{N}_{p,k} + 1$ to $\hat{N}_{p,k} + J_p$ **do**
    proposal: $\mathbf{x}_k^{(i)} \sim \gamma_{1,k}(\mathbf{x}), \quad w_{k|k-1}^{(i)} = \frac{p_{b,k}(\mathbf{x}_k^{(i)})}{J_p \gamma_{1,k}(\mathbf{x}_k^{(i)})}$
  end for
  $\hat{N}_{k|k-1} = \sum_j w_{k|k-1}^{(j)}$, $\hat{\mathbf{m}}_k = \frac{1}{\hat{N}_{k|k-1}} \sum_j w_{k|k-1}^{(j)} g(\mathbf{x}_k^{(j)})$, $\hat{\Sigma}_k = \frac{1}{\hat{N}_{k|k-1}} \sum_j w_{k|k-1}^{(j)} g(\mathbf{x}_k^{(j)}) g^T(\mathbf{x}_k^{(j)})$
  for $i = 1$ to $\hat{N}_{p,k} + J_p$ **do**
    weight update: $w_k^{(i)} = w_{k|k-1}^{(i)} \times \frac{\mathcal{N}_{\Sigma_{\mathbf{z}} + \Sigma_k^o}(\mathbf{z}_k - g(\mathbf{x}_k^{(i)}) - \mathbf{m}_k^o)}{\mathcal{N}_{\Sigma_{\mathbf{z}} + \Sigma_k}(\mathbf{z}_k - \mathbf{m}_k)}$
  end for
  for $i = \hat{N}_{p,k} + 1$ to $\hat{N}_{p,k} + J_p$ **do**
    proposal: $\mathbf{x}_k^{(i)} \sim \gamma_{2,k}(\mathbf{x}), \quad w_{k|k-1}^{(i)} = \frac{p_{b,k}(\mathbf{x}_k^{(i)})}{J_p \gamma_{2,k}(\mathbf{x}_k^{(i)})}$
  end for
  Normalize $\{w_{k|k-1}^{(i)}\}_{i=\hat{N}_{p,k}+1}^{i=\hat{N}_{p,k}+J_p}$
  Weight update and state estimation
  $\hat{N}_{k|k-1} = \sum_j w_{k|k-1}^{(j)}$, $\hat{\mathbf{m}}_k = \frac{1}{\hat{N}_{k|k-1}} \sum_j w_{k|k-1}^{(j)} g(\mathbf{x}_k^{(j)})$, $\hat{\Sigma}_k = \frac{1}{\hat{N}_{k|k-1}} \sum_j w_{k|k-1}^{(j)} g(\mathbf{x}_k^{(j)}) g^T(\mathbf{x}_k^{(j)})$
  for $i = 1$ to $\hat{N}_{p,k} + J_p$ **do**
    weight update: $w_k^{(i)} = w_{k|k-1}^{(i)} \times \frac{\mathcal{N}_{\Sigma_{\mathbf{z}} + \Sigma_k^o}(\mathbf{z}_k - g(\mathbf{x}_k^{(i)}) - \mathbf{m}_k^o)}{\mathcal{N}_{\Sigma_{\mathbf{z}} + \Sigma_k}(\mathbf{z}_k - \mathbf{m}_k)}$
  end for
  cardinality update: $\pi_{k|k}(n) = K_c^{-1} \pi_{k|k-1}(n) \times \frac{\mathcal{N}_{\Sigma_{\mathbf{z}} + \Sigma_k^n}(\mathbf{z}_k - n\hat{\mathbf{m}}_k)}{\mathcal{N}_{\Sigma_{\mathbf{z}} + \Sigma_k}(\mathbf{z}_k - \mathbf{m}_k)}$
  target number estimation: $\hat{N}_k = \mathrm{MAP}(\pi_{k|k}(n))$
  resample step: $\{w_k^{(i)}, \mathbf{x}_k^{(i)}\}_{i=1}^{i=\hat{N}_{p,k}+J_p} \rightarrow \{\mathbf{x}_k^{(i)}\}_{i=1}^{i=\hat{N}_k \times N_{ppt}}$
  clustering step: $\{\hat{\mathbf{x}}_{k,n}\}_{n=1}^{\hat{N}_k} = \mathrm{cluster}(\{\mathbf{x}_k^{(i)}\}_{i=1}^{i=\hat{N}_k \times N_{ppt}}, \hat{N}_k)$
end for

Fig. 4.2 Auxiliary particle filter implementation of the approximate CPHD filter.

targets is N_0, then the combined complexity of these operations is $\mathcal{O}(N_0^2 N_p^2)$. Thus the overall complexity for one iteration of PHD filter is $\mathcal{O}(N_p M^2 + M^3 + N_0^2 N_p^2)$.

The CPHD filter additionally propagates the cardinality, but the computational requirements are minor. The CPHD weight update equations require computing multiple covariance matrices and their inverse, which is $\mathcal{O}(N_p M^2 + M^3)$. The CPHD cardinality update involves multiple matrix inversion and is $\mathcal{O}(N_0 M^3)$. Clustering is performed only once using the k-means algorithm and has computational complexity $\mathcal{O}(N_0 N_p)$. The overall complexity for one iteration of the CPHD filter is $\mathcal{O}(N_p M^2 + N_0 M^3 + N_0 N_p)$. From the expressions it can be seen that the CPHD filter computation is dominated by matrix inversions and clustering, whereas the PHD filter has additional computational requirements for estimation of the number of targets.

4.5 Numerical simulations

In this section we compare the PHD and CPHD filters using numerical simulations of multitarget tracking in superpositional sensor environments. Specifically we compare the approximate PHD and CPHD filter implementations discussed in Section 4.4 with an MCMC filter [3] (Section 3.3.3) that tracks the joint marginal posterior over time.

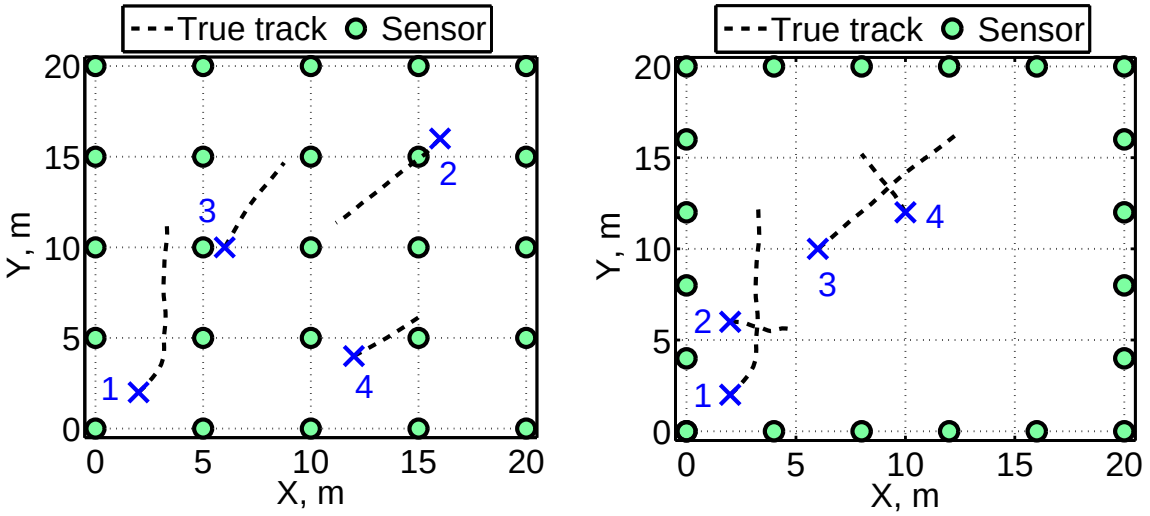
4.5.1 Target dynamics

For each target its dynamics are assumed to be independent of the other targets and their dynamics. We represent the state of object i at time k, $\mathbf{x}_{k,i}$, by a four-dimensional vector: position on the x-axis and y-axis, velocity on the x-axis and y-axis. The targets move within the boundaries of the monitoring area according to the discrete nearly constant velocity model [158]

$$\mathbf{x}_{k+1,i} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{k,i} + \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ T & 0 \\ 0 & T \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix} \tag{4.73}$$

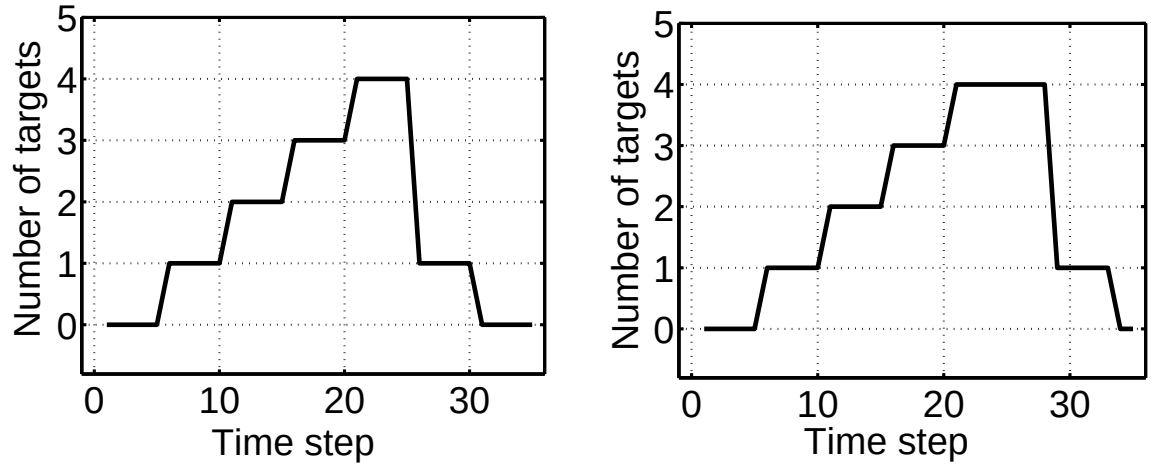
where T is the sampling period and u_x, u_y are zero-mean Gaussian white noise with respective variance $\sigma_{u_x}^2$ and $\sigma_{u_y}^2$. Targets can randomly appear and disappear within the

monitored region. For simulating the reference target motion, the model parameters are set to $T = 0.25s$, $\sigma^2_{u_x} = \sigma^2_{u_y} = 0.35$.



(a) Tracks 1 (b) Tracks 2

Fig. 4.3 Two sets of tracks (a) Tracks 1 (b) Tracks 2 used in the simulations.



(a) Tracks 1 (b) Tracks 2

Fig. 4.4 Plot of number of targets vs. time: (a) Tracks 1 (b) Tracks 2.

Two sets of target tracks are used in the simulations as shown in Figures 4.3(a) and 4.3(b) and are henceforth referred to as Tracks 1 and Tracks 2 respectively. The simulation is

run for 35 time steps covering a total duration of $35 \times 0.25 = 8.75s$. For these tracks the evolution of target number over time is given in Figures 4.4(a) and 4.4(b). In Tracks 1, all the targets are well separated in space at all times. In Tracks 2, two or more targets are in close vicinity for some parts of the simulation. Specifically, the targets 1 and 2 and the targets 3 and 4 approach each other and then diverge. Figure 4.5 illustrates the proximity of the target pairs, showing the Euclidean distance as a function of time for the target pairs 1, 2 and 3, 4.

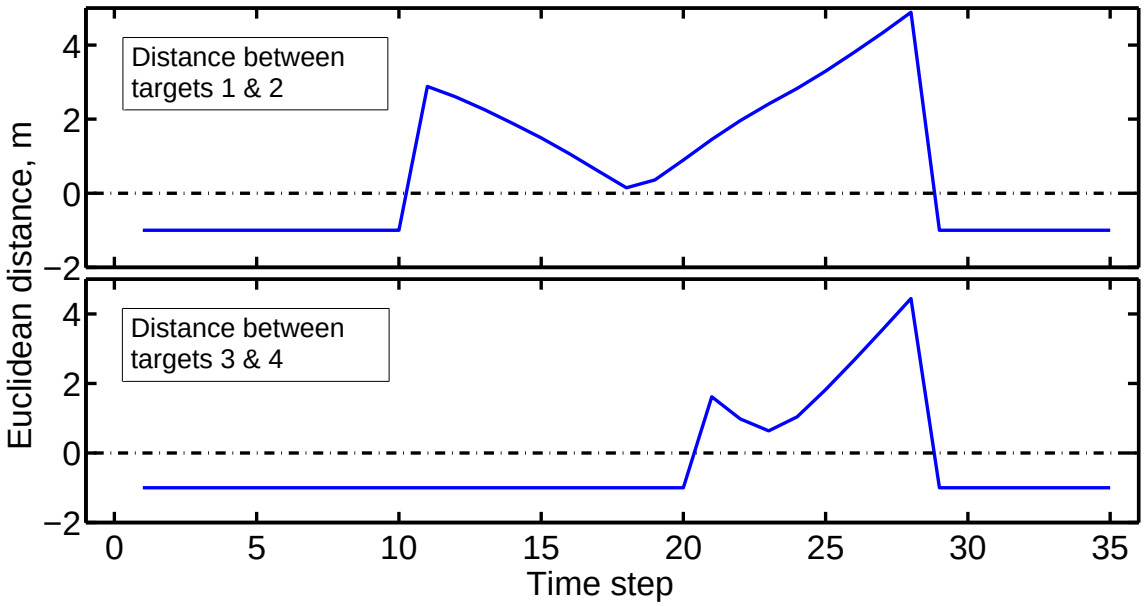


Fig. 4.5 Plot of Euclidean distance vs. time for target pairs 1,2 and 3,4 in Tracks 2. When either target in the pair is absent, the distance is indicated as -1.

4.5.2 Algorithm settings

We now discuss the algorithm implementation choices. For practical purposes we need to assume a limit on the maximum number of targets that could be present at any give time. This limit can be chosen much higher than the true number of targets. We use a maximum of $N_0 = 6$ targets for the PHD and the MCMC filter and $N_0 = 9$ targets for the CPHD filter. A smaller value is used for PHD and MCMC filters as it significantly affects the computational time required for their processing.

We use the Silhouette method [154] to estimate the number of clusters for the PHD filter. The number of clusters is varied from 2 to 6 and the choice which maximizes the average silhouette gives the estimate of the number of targets present. The method cannot identify when only a single target is present. The centroids of the clusters are the target state estimates. Identifying the number of targets in the CPHD filter implementation is much easier and maximum a posteriori (MAP) estimation is used. The peak in the cardinality distribution provides the target number estimate and computational requirements are minimal. The k-means clustering technique is then applied to group the particles into clusters and the centroids of the clusters are the target state estimates. For the MCMC filter the single target state $\mathbf{x}_{k,i}$ is extended to include a binary indicator variable $e_{k,i}$ which indicates the presence or absence of the target. A target is said to be present if more than half of the corresponding target particles have their indicator variable set to one. The mean of the particles are used as target state estimates.

The filtering algorithms assume that the probability of survival of existing targets to be uniform throughout the monitoring region with $p_{sv} = 0.9$. For the PHD filter the birth process is assumed to be Poisson with mean cardinality of 0.2. For the CPHD filter the birth process is assumed to be IIDC with cardinality distribution $p_b(n)$ to be Geometric with parameter 0.2. The target birth intensity $b_{k+1|k}(\mathbf{x})$ is uniform throughout the monitoring region for position component of target state for both PHD and CPHD filters. For the MCMC filter we assume that a new target can be born anywhere in the monitoring region with uniform probability and the probability of birth of a new target is 0.2. The two velocity components of the new particles are initialized using a standard normal distribution $\mathcal{N}(0, 1)$.

The number of particles per target is set to $N_{ppt} = 500$ and the number of new particles added to account for the birth process is $J_p = N_{ppt}$. Increasing N_{ppt} or J_p beyond 500 did not give any significant improvement in the accuracy of filter performance. In the auxiliary implementation of filters, particles for the birth process are drawn from the auxiliary distribution with probability $p = 0.9$. Since direct initialization of birth process particles using the measurements is difficult for superpositional observation model, a high value of p is used. A Gaussian jitter covariance of $\Sigma_v = \sigma_v^2 \cdot I_{2\times 2}$, where $\sigma_v^2 = 0.25$ is used to keep the particle set sufficiently diverse. For the MCMC filter the burn-in is 1000 and the thinning factor is 3 which are standard values from the literature, observed to be sufficient in many cases to substantially reduce correlation between samples. The OSPA metric (Section 2.7)

is used to calculate the error between the true and estimated multitarget sets.

The simulations are repeated multiple times with different random initializations and the average OSPA error is reported over all the simulations. The target trajectory is the same for each random initialization. A set of 20 different measurements are generated and each is processed with 5 different random initializations for all the algorithms. Thus the average error is reported over $20 \times 5 = 100$ simulations in order to reduce the variability introduced due to the stochastic nature of processing.

4.5.3 Acoustic amplitude sensors

The acoustic sensor likelihood model is discussed in Section 2.4.2. The moving targets are monitored by 25 acoustic sensors distributed in a uniform grid. The targets emit a signal which has amplitude $A = 10$ at unit distance from the target. The sensors have a path loss exponent of $\kappa = 1$. When the targets lie within $d_0 = 0.2m$ distance of any sensor, the sensors record the same amplitude of A/d_0. This avoids any singularities in the measurements. The sensors are assumed to have a Gaussian noise variance of $\sigma_{\mathbf{z}}^2 = 0.05$.

The target tracks of Tracks 1 are used in this simulation. Table 4.1 presents the average error over 100 random initializations. The methods of CPHD, PHD and MCMC are used for tracking. The error values are reported for different values of cardinality penalty factor ($c = 1, 2.5, 5$). The CPHD filter has the lowest OSPA error at all values of c indicating very few cardinality errors and accurate target location estimates.

Tracks 1	OSPA error		
Algorithm	$c = 1$	$c = 2.5$	$c = 5$
CPHD	0.34	0.44	0.47
PHD	0.71	1.44	2.61
MCMC	0.50	0.80	0.99

Table 4.1 Acoustic amplitude sensors: average OSPA error.

Figure 4.6 shows the box-and-whisker plot of the error over time for the various methods. The PHD filter has a high error when the number of targets is one because the Silhouette method used to find the number of clusters from the particles cannot estimate a single cluster. The accurate cardinality prediction using the CPHD filter is able to effectively mitigate this problem. Figures 4.7(a) and 4.7(b) show the true target trajectories and estimated target locations as obtained using the CPHD and MCMC filters.

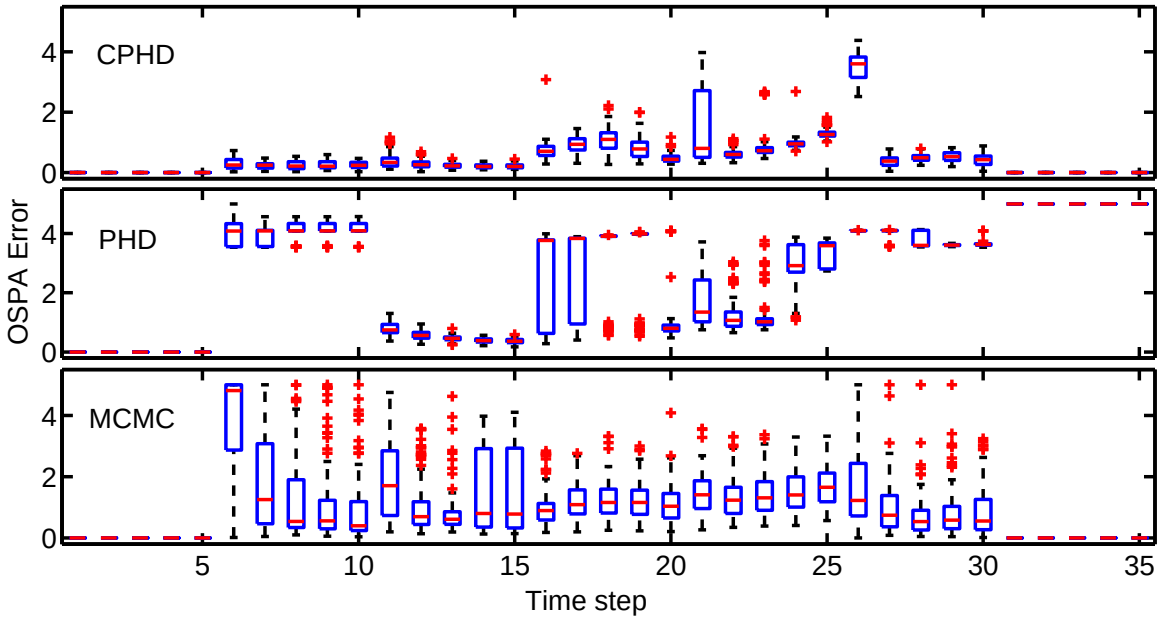
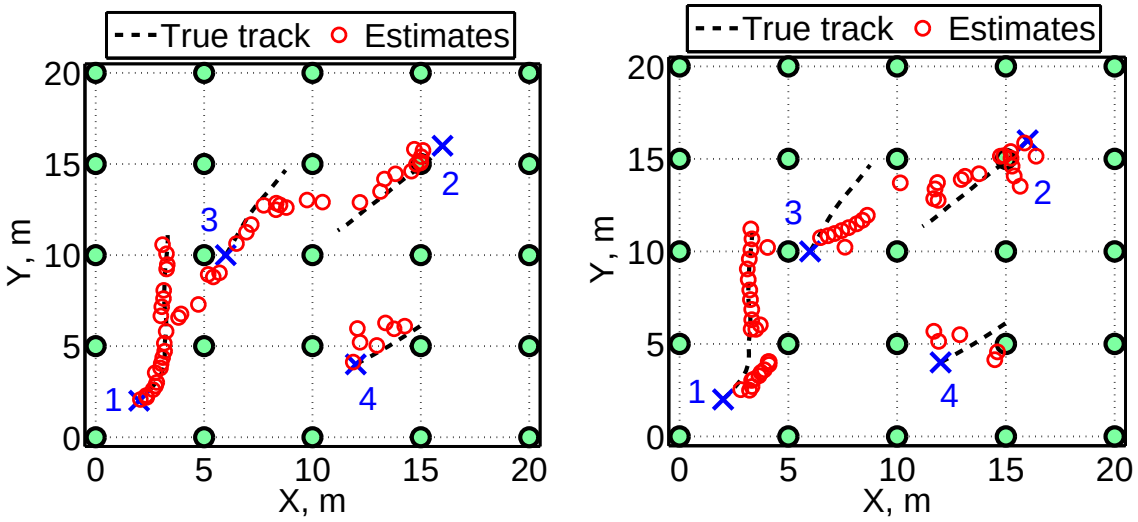


Fig. 4.6 Acoustic amplitude sensors: Box-and-whisker plot of the error over time for the CPHD, PHD and MCMC methods with $c = 5$. Boxes indicate 25-75 interquartile range; whiskers extend 1.5 times the range and '+' symbols indicate outliers lying beyond the whiskers.



(a) CPHD (b) MCMC

Fig. 4.7 Acoustic amplitude sensors: True target tracks and target location estimates (circles) obtained using the CPHD and MCMC methods.

4.5.4 Radio frequency tomography

The radio frequency tomography sensor system is described in Section 3.1 and the measurement model we simulate is discussed in Section 3.2.1. 24 radio frequency sensors are placed on the periphery of the monitoring region to form a sensor network. The 24 sensors give rise to a total of 276 unique bidirectional links. The measurement model parameters are $\phi = 5$ and $\sigma_\lambda = 0.2$. These parameter values are based on the empirical values observed from experiments (Chapter 3). The Gaussian measurement noise has variance $\sigma_{\mathbf{z}}^2 = 0.25$.

Track. 1	OSPA error		
Algorithm	$c = 1$	$c = 2.5$	$c = 5$
CPHD	0.16	0.20	0.23
PHD	0.57	1.29	2.48
MCMC	0.34	0.43	0.48

Table 4.2 Radio-frequency sensors: Average OSPA error.

We use the RF tomography sensor system simulations to analyse the performance of filters using Tracks 2. The simulated observations are used to track the targets using the CPHD, PHD and MCMC algorithms. A summary of the average OSPA error, performed over 100 random simulations, is provided in Table 4.2. The error values are reported for different values of cardinality error penalty ($c = 1, 2.5, 5$). Overlapping trajectories and closely-spaced targets lead to higher average errors for all the algorithms, but the measurement dimension and the signal-to-noise ratio are much higher for the RF tomography setup, so the average errors are smaller than in the acoustic sensor case.

A detailed error behaviour over time can be seen from the box and whisker plot in the Figure 4.8. At time = 6 we observe that the MCMC filter has a much higher error median indicating difficulty in identifying the appearance of first target within the network. Also since tracking in the joint target state domain is difficult the median error at subsequent times is higher when compared with the CPHD filter. Figures 4.9(a) and 4.9(b) plot example trajectory estimates using the different algorithms for the case of crossing targets.

4.5.5 Computational requirements

Table 4.3 summarizes the computational time required for processing a single observation vector for each of the algorithms. All the simulations were performed using algorithms

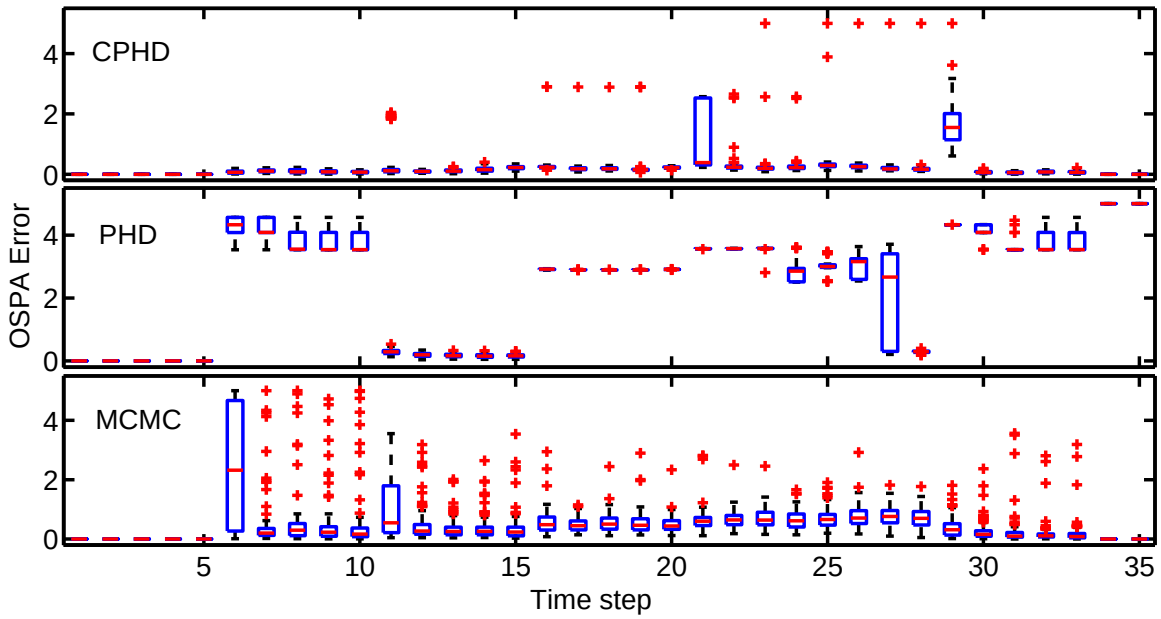
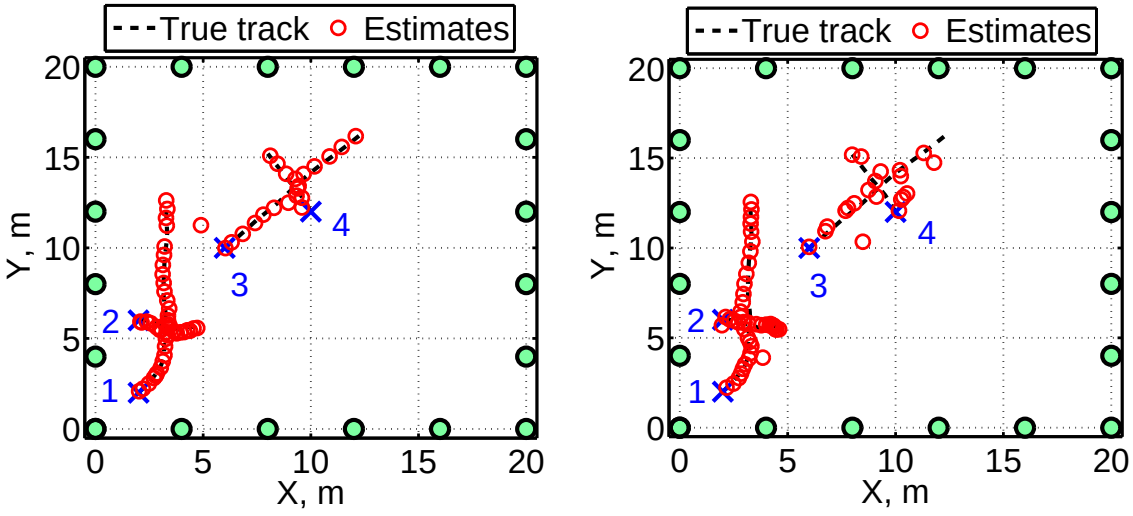


Fig. 4.8 Radio-frequency sensors: Box-and-whisker plot of error over time for the methods of CPHD, PHD and MCMC with $c = 5$ for overlapping target trajectories. Boxes indicate 25-75 interquartile range; whiskers extend 1.5 times the range and '+' symbols indicate outliers lying beyond the whiskers.



(a) CPHD (b) MCMC

Fig. 4.9 Radio-frequency sensors: True target tracks and the target location estimates (circles) obtained using the CPHD and MCMC methods.

implemented in Matlab running on computers with two Xeon 4-core 2.5GHz processors and 14GB RAM. The CPHD filter is the fastest and also the most accurate filter. A significant portion of the PHD filter computational time is spent towards identifying the number of targets. This could certainly be reduced substantially by adoption of an alternative technique, but in light of the good performance of the CPHD filter, we have not been motivated to conduct further exploration beyond the examination of various methods that was conducted in [157]. The MCMC filter is the slowest owing to the sequential nature of the algorithm and the fact that it operates in multitarget state space.

Algorithm	Acoustic sensors	RF Tomography
CPHD	0.12 ± 0.01	0.70 ± 0.02
PHD	3.11 ± 0.19	9.73 ± 0.47
MCMC	10.01 ± 0.07	22.13 ± 0.18

Table 4.3 CPU time required in seconds for different algorithms.

4.6 Conclusion

We derived computationally-tractable approximations of the PHD and CPHD filters for superpositional sensors in this chapter. The key steps in the filter derivations are the application of a change of variables formula and Campbell's theorem. The former allows us to shift our analysis from random sets to random variables; the latter allows us to express the first and second moments of the transformed random variables using the PHD and the second factorial moment of the multitarget distribution. We proposed auxiliary particle filter based implementations of the approximate filters and performed a simulation-based analysis of the filters using models of acoustic amplitude sensors and radio-frequency tomography sensor systems. The CPHD filter accurately tracks the target locations and the number of targets, significantly outperforming the PHD filter which suffers from an unreliable cardinality estimate. It also outperforms a more computationally demanding joint-posterior MCMC filter. In common with most particle-based implementations of PHD/CPHD filters, the algorithms presented here rely on a clustering procedure to form a final estimate of the target states. This limitation motivates further investigation into more elegant solutions. Towards this direction, multi-Bernoulli filters for superpositional sensors will be discussed in the next chapter.

Chapter 5

Multi-Bernoulli filters for superpositional sensors

Chapter 4 discussed the PHD and CPHD filters for superpositional sensors. Though the CPHD filter performs accurate multitarget tracking it is inherently limited by the fact that a single density function is used to represent the states of multiple targets. The accuracy of multitarget tracking can be potentially increased by using one density function to represent each target. This can be achieved by modeling the multitarget state as realization of a multi-Bernoulli random finite set. A multi-Bernoulli RFS is the union of multiple independent Bernoulli random finite sets each of which is characterized using a probability of existence and a state density function.

Motivated by this reasoning, in this chapter we develop multi-Bernoulli filters for superpositional sensors. The novel contributions of the thesis in this chapter are

- derivation of approximate update equations for the multi-Bernoulli filter for superpositional sensors,
- derivation of approximate update equations for the hybrid multi-Bernoulli CPHD filter for superpositional sensors,
- development of auxiliary particle filter implementations of the multi-Bernoulli filter and hybrid multi-Bernoulli CPHD filter.

The first filter we derive models the target birth process and the multitarget state of existing targets as multi-Bernoulli random finite sets and is called the multi-Bernoulli filter.

This filter can suffer from difficulties in accurately identifying the the number of targets. To overcome this we propose a second filter which models the target birth process as independent and identically distributed cluster random finite set and the multitarget state of existing targets as a multi-Bernoulli random finite set and is called the hybrid multi-Bernoulli CPHD filter. It makes use of the accurate cardinality representation provided by IIDC process to correctly identify the target number.

This chapter is organized as follows. In Section 5.1, we discuss an approximate PHD update mechanism for a union of independent random finite sets. As special cases we develop the multi-Bernoulli and the hybrid multi-Bernoulli CPHD filter update equations in Section 5.2 and Section 5.3, respectively. The auxiliary particle filter implementation of these filters is presented in Section 5.4. Section 5.5 compares these filters with the CPHD filter in the context of multitarget tracking using acoustic sensor networks and the radio frequency (RF) tomography measurement model.

5.1 Union of independent random finite sets

Consider an RFS Ξ which is the union of multiple independent random finite sets. Let $\Xi = \Xi^A \cup \Xi^B \cup \Xi^C \cup \cdots$ and the PHD function for each of the individual RFS components be denoted by $D^A(\mathbf{x}), D^B(\mathbf{x}), D^C(\mathbf{x}), \cdots$ respectively. Let the PHD of the RFS Ξ be denoted $D(\mathbf{x})$. Then we can show that

$$D(\mathbf{x}) = D^A(\mathbf{x}) + D^B(\mathbf{x}) + D^C(\mathbf{x}) + \cdots \cdot \tag{5.1}$$

The above result can be easily proved using the properties of probability-generating functional and the basic rules for functional derivatives (Chap. 11, [34]).

Let $f^A(W)$ and $f^B(W)$ be the multitarget probability densities of two independent random finite sets Ξ^A and Ξ^B respectively. Let $f(W)$ be the multitarget probability density of $\Xi = \Xi^A \cup \Xi^B$. Then we have the following convolution relation between the densities (Sec. 11.5.3, Chap. 11, [34])

$$f(W) = \sum_{Y \subseteq W} f^A(Y) f^B(W \setminus Y) \tag{5.2}$$

where the summation is over all subsets Y of W. The set $(W \setminus Y)$ is the difference set

which includes elements from W not present in Y. Note that we can express densities of random finite sets consisting of more than two independent random finite sets using a similar convolution relation.

5.1.1 PHD update for the union of independent random finite sets

In this section we analyze the PHD update step when the multitarget state can be expressed as union of multiple independent random finite sets. We limit our discussion to the case of the union of two independent random finite sets but it can be easily extended to the case where more than two independent random finite sets are present.

Let $\mathcal{X}$ denote the single target state space and $\mathcal{Y}$ denote the space of observations. Let the predicted multitarget state at time $k+1$ be modeled as a random finite set Ξ with density $f_{k+1|k}(W)$ and PHD $D_{k+1|k}(\mathbf{x})$. Further assume that Ξ is the union of independent random finite sets $\Xi^A_{k+1|k}$ and $\Xi^B_{k+1|k}$ with densities $f^A_{k+1|k}(Y)$ and $f^B_{k+1|k}(Y)$ respectively and PHDs $D^A_{k+1|k}(\mathbf{x})$ and $D^B_{k+1|k}(\mathbf{x})$ respectively. Then we have

$$f_{k+1|k}(W) = \sum_{Y \subseteq W} f^A_{k+1|k}(Y) f^B_{k+1|k}(W \setminus Y) \tag{5.3}$$

$$D_{k+1|k}(\mathbf{x}) = D^A_{k+1|k}(\mathbf{x}) + D^B_{k+1|k}(\mathbf{x}). \tag{5.4}$$

Let $f_{k+1|k+1}(W)$ and $D_{k+1|k+1}(\mathbf{x})$ be the density and PHD of the posterior multitarget state. Then we have

$$D_{k+1|k+1}(\mathbf{x}) = \int_{\mathcal{X}} f_{k+1|k+1}(\{\mathbf{x}\} \cup W) \delta W. \tag{5.5}$$

Let $\mathbf{z}_{k+1}$ be the measurement vector at time $k+1$ and let $h_{\mathbf{z}_{k+1}}(W)$ be the multitarget likelihood function. Applying Bayes rule we get

$$D_{k+1|k+1}(\mathbf{x}) = \frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f_{k+1|k}(\{\mathbf{x}\} \cup W)\, \delta W}{f_{k+1}(\mathbf{z}_{k+1}|Z_{1:k})}. \tag{5.6}$$

From (5.3) we have

$$f_{k+1|k}(\{\mathbf{x}\} \cup W) = \sum_{Y \subseteq W} f^{A}_{k+1|k}(\{\mathbf{x}\} \cup Y) f^{B}_{k+1|k}(W \setminus Y) + \sum_{Y \subseteq W} f^{B}_{k+1|k}(\{\mathbf{x}\} \cup Y) f^{A}_{k+1|k}(W \setminus Y) \tag{5.7}$$

$$= D^{A}_{k+1|k}(\mathbf{x}) \sum_{Y \subseteq W} \frac{f^{A}_{k+1|k}(\{\mathbf{x}\} \cup Y)}{D^{A}_{k+1|k}(\mathbf{x})} f^{B}_{k+1|k}(W \setminus Y) + D^{B}_{k+1|k}(\mathbf{x}) \sum_{Y \subseteq W} \frac{f^{B}_{k+1|k}(\{\mathbf{x}\} \cup Y)}{D^{B}_{k+1|k}(\mathbf{x})} f^{A}_{k+1|k}(W \setminus Y) \tag{5.8}$$

$$= D^{A}_{k+1|k}(\mathbf{x}) \sum_{Y \subseteq W} f^{A_{\mathbf{x}}}_{k+1|k}(Y) f^{B}_{k+1|k}(W \setminus Y) + D^{B}_{k+1|k}(\mathbf{x}) \sum_{Y \subseteq W} f^{B_{\mathbf{x}}}_{k+1|k}(Y) f^{A}_{k+1|k}(W \setminus Y) \tag{5.9}$$

where $f^{A_{\mathbf{x}}}_{k+1|k}(Y)$ and $f^{B_{\mathbf{x}}}_{k+1|k}(Y)$ are defined as

$$f^{A_{\mathbf{x}}}_{k+1|k}(Y) = \frac{f^{A}_{k+1|k}(\{\mathbf{x}\} \cup Y)}{D^{A}_{k+1|k}(\mathbf{x})}, \tag{5.10}$$

$$f^{B_{\mathbf{x}}}_{k+1|k}(Y) = \frac{f^{B}_{k+1|k}(\{\mathbf{x}\} \cup Y)}{D^{B}_{k+1|k}(\mathbf{x})}. \tag{5.11}$$

These are valid multitarget densities which integrate to 1 from the definition of PHD in Section 2.2.3. Now denote

$$f^{A^{*}_{\mathbf{x}}}_{k+1|k}(W) = \sum_{Y \subseteq W} f^{A_{\mathbf{x}}}_{k+1|k}(Y) f^{B}_{k+1|k}(W \setminus Y) \tag{5.12}$$

$$f^{B^{*}_{\mathbf{x}}}_{k+1|k}(W) = \sum_{Y \subseteq W} f^{B_{\mathbf{x}}}_{k+1|k}(Y) f^{A}_{k+1|k}(W \setminus Y) \tag{5.13}$$

which are valid multitarget densities. We note that the densities $f^{A^{*}_{\mathbf{x}}}_{k+1|k}(W)$ and $f^{B^{*}_{\mathbf{x}}}_{k+1|k}(W)$ are obtained by replacing $f^{A}_{k+1|k}(Y)$ and $f^{B}_{k+1|k}(Y)$ with $f^{A_{\mathbf{x}}}_{k+1|k}(Y)$ and $f^{B_{\mathbf{x}}}_{k+1|k}(Y)$ respectively in (5.3). The density $f^{A^{*}_{\mathbf{x}}}_{k+1|k}(W)$ can be interpreted as the multitarget distribution of the RFS which is the union of independent random finite sets with multitarget densities given by $f^{A_{\mathbf{x}}}_{k+1|k}(Y)$ and $f^{B}_{k+1|k}(Y)$. In general, we denote by $f^{A^{*}_{\mathbf{x}}}_{k+1|k}(W)$ the multitarget density obtained by replacing $f^{A}_{k+1|k}(Y)$ with $f^{A_{\mathbf{x}}}_{k+1|k}(Y)$ in the convolution expression for $f_{k+1|k}(W)$.

This notation allows us to interpret the density $f^{A^*_{\mathbf{x}}}_{k+1|k}(W)$ in general when more than two independent RFS components are involved. Using this notation we can write,

$$f_{k+1|k}(\{\mathbf{x}\} \cup W) = D^A_{k+1|k}(\mathbf{x}) f^{A^*_{\mathbf{x}}}_{k+1|k}(W) + D^B_{k+1|k}(\mathbf{x}) f^{B^*_{\mathbf{x}}}_{k+1|k}(W). \tag{5.14}$$

Substituting (5.14) in (5.6)

$$D_{k+1|k+1}(\mathbf{x}) = D^A_{k+1|k}(\mathbf{x}) \frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f^{A^*_{\mathbf{x}}}_{k+1|k}(W)\, \delta W}{f_{k+1}(\mathbf{z}_{k+1}|Z_{1:k})} + D^B_{k+1|k}(\mathbf{x}) \frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f^{B^*_{\mathbf{x}}}_{k+1|k}(W)\, \delta W}{f_{k+1}(\mathbf{z}_{k+1}|Z_{1:k})}. \tag{5.15}$$

Now define

$$\mathcal{D}^A_{k+1|k+1}(\mathbf{x}) = D^A_{k+1|k}(\mathbf{x}) \frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f^{A^*_{\mathbf{x}}}_{k+1|k}(W)\, \delta W}{f_{k+1}(\mathbf{z}_{k+1}|Z_{1:k})} \tag{5.16}$$

$$\mathcal{D}^B_{k+1|k+1}(\mathbf{x}) = D^B_{k+1|k}(\mathbf{x}) \frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f^{B^*_{\mathbf{x}}}_{k+1|k}(W)\, \delta W}{f_{k+1}(\mathbf{z}_{k+1}|Z_{1:k})}. \tag{5.17}$$

We now assume that the posterior multitarget state at time $k+1$ is a union of independent random finite sets Ξ^A_{k+1} and Ξ^B_{k+1}. Also let $D^A_{k+1|k+1}(\mathbf{x})$ and $D^B_{k+1|k+1}(\mathbf{x})$ be their PHD functions. Then from (5.15), (5.16) and (5.17) we have

$$D^A_{k+1|k+1}(\mathbf{x}) + D^B_{k+1|k+1}(\mathbf{x}) = \mathcal{D}^A_{k+1|k+1}(\mathbf{x}) + \mathcal{D}^B_{k+1|k+1}(\mathbf{x}). \tag{5.18}$$

To derive an update mechanism for propagating the PHD over time, we make the following separability assumption on the PHD of the different components

$$D^A_{k+1|k+1}(\mathbf{x}) \approx \mathcal{D}^A_{k+1|k+1}(\mathbf{x}) \tag{5.19}$$

$$D^B_{k+1|k+1}(\mathbf{x}) \approx \mathcal{D}^B_{k+1|k+1}(\mathbf{x}). \tag{5.20}$$

This one-to-one matching is an approximation and is based on the assumption that the posterior multitarget state is also a union of independent random finite sets and that for each component in the predicted multitarget state we have a corresponding component of the same type in the posterior multitarget state. The above approximation is justified

under the assumption that the supports of the PHD functions of the two components are well separated. This is a good approximation if each component represents a single target or a group of targets which are well separated in the state space.

This approximation allows us to relate the posterior and predicted PHD functions, using (5.16) and (5.19), as follows

$$D^{A}_{k+1|k+1}(\mathbf{x}) \approx D^{A}_{k+1|k}(\mathbf{x}) \frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f^{A^{*}_{\mathbf{x}}}_{k+1|k}(W)\, \delta W}{f_{k+1}(\mathbf{z}_{k+1}|Z_{1:k})}. \tag{5.21}$$

In general we can update the PHD of each individual RFS component of the multitarget state using the above approximation. We apply this PHD update mechanism for two specific cases: (i) when the multitarget state is modelled as the union of independent Bernoulli RFS components, leading to the multi-Bernoulli filter; and (ii) when the multitarget state is modelled as union of independent multi-Bernoulli RFS and IIDC RFS components, leading to the hybrid multi-Bernoulli CPHD filter.

5.1.2 Approximate PHD update for the superpositional sensor model with Gaussian sensor noise

In this section we further approximate the update equation in (5.21) for the case of the superpositional sensor model under the assumption of Gaussian sensor noise. This approximation leads to a computationally tractable update equation for the conditional PHD. This result is applied for deriving the update equations for the multi-Bernoulli filter and the hybrid multi-Bernoulli CPHD filter in later sections.

From (5.16) we have

$$\mathcal{D}^{A}_{k+1|k+1}(\mathbf{x}) = D^{A}_{k+1|k}(\mathbf{x}) \frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f^{A^{*}_{\mathbf{x}}}_{k+1|k}(W)\, \delta W}{f_{k+1}(\mathbf{z}_{k+1}|Z_{1:k})} \tag{5.22}$$

$$= D^{A}_{k+1|k}(\mathbf{x}) \frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(\{\mathbf{x}\} \cup W) f^{A^{*}_{\mathbf{x}}}_{k+1|k}(W)\, \delta W}{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(W) f_{k+1|k}(W)\, \delta W} \tag{5.23}$$

Using the Gaussian sensor noise assumption and the superpositional likelihood model from

Section 2.4.2, we have:

$$\mathcal{D}^{A}_{k+1|k+1}(\mathbf{x}) = D^{A}_{k+1|k}(\mathbf{x}) \frac{\int_{\mathcal{X}} \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \zeta(W))\, f^{A^*_{\mathbf{x}}}_{k+1|k}(W)\, \delta W}{\int_{\mathcal{X}} \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - \zeta(W))\, f_{k+1|k}(W)\, \delta W} \tag{5.24}$$

We apply the transformation $\mathbf{y}^* = \zeta(W)$ in the numerator and $\mathbf{y} = \zeta(W)$ in the denominator. Using the formula for change of variables for set integrals from Section 2.2.4 we have,

$$\mathcal{D}^{A}_{k+1|k+1}(\mathbf{x}) = D^{A}_{k+1|k}(\mathbf{x}) \frac{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \mathbf{y}^*)\, Q^{A^*_{\mathbf{x}}}_{k+1|k}(\mathbf{y}^*)\, d\mathbf{y}^*}{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - \mathbf{y})\, Q_{k+1|k}(\mathbf{y})\, d\mathbf{y}}, \tag{5.25}$$

where $Q_{k+1|k}(\mathbf{y})$ and $Q^{A^*_{\mathbf{x}}}_{k+1|k}(\mathbf{y}^*)$ are the probability distributions of the random vectors $\mathbf{y}$ and $\mathbf{y}^*$ respectively. Using a Gaussian approximation for these densities,

$$Q_{k+1|k}(\mathbf{y}) \approx \mathcal{N}_{\Sigma_{k+1}}(\mathbf{y} - \mathbf{m}_{k+1}) \tag{5.26}$$

$$Q^{A^*_{\mathbf{x}}}_{k+1|k}(\mathbf{y}^*) \approx \mathcal{N}_{\Sigma^{A^*_{\mathbf{x}}}_{k+1}}(\mathbf{y}^* - \mathbf{m}^{A^*_{\mathbf{x}}}_{k+1}), \tag{5.27}$$

we have

$$\mathcal{D}^{A}_{k+1|k+1}(\mathbf{x}) \approx D^{A}_{k+1|k}(\mathbf{x}) \frac{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \mathbf{y}^*)\, \mathcal{N}_{\Sigma^{A^*_{\mathbf{x}}}_{k+1}}(\mathbf{y}^* - \mathbf{m}^{A^*_{\mathbf{x}}}_{k+1})\, d\mathbf{y}^*}{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_{\mathbf{z}}}(\mathbf{z}_{k+1} - \mathbf{y})\, \mathcal{N}_{\Sigma_{k+1}}(\mathbf{y} - \mathbf{m}_{k+1})\, d\mathbf{y}}. \tag{5.28}$$

The above equation can be simplified using the result in (2.55) from Section 2.6. Combining the approximations in (5.21) and (5.28), the approximate PHD update equation for the superpositional sensor model with Gaussian sensor noise is

$$D^{A}_{k+1|k+1}(\mathbf{x}) \approx D^{A}_{k+1|k}(\mathbf{x}) \frac{\mathcal{N}_{\Sigma_{\mathbf{z}} + \Sigma^{A^*_{\mathbf{x}}}_{k+1}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \mathbf{m}^{A^*_{\mathbf{x}}}_{k+1})}{\mathcal{N}_{\Sigma_{\mathbf{z}} + \Sigma_{k+1}}(\mathbf{z}_{k+1} - \mathbf{m}_{k+1})} \tag{5.29}$$

where $\mathbf{m}_{k+1}$ and Σ_{k+1} are the mean and covariance matrix of the distribution $Q_{k+1|k}(\mathbf{y})$ and $\mathbf{m}^{A^*_{\mathbf{x}}}_{k+1}$ and $\Sigma^{A^*_{\mathbf{x}}}_{k+1}$ are the mean and covariance matrix of the distribution $Q^{A^*_{\mathbf{x}}}_{k+1|k}(\mathbf{y}^*)$. These mean and covariance matrix parameters can be found using the quadratic version of Campbell's theorem (Section 2.2.5). By modeling the unknown multitarget state as a union of statistically independent random finite sets, different tracking filters can be derived

whose update equations are special cases of (5.29).

A brief analysis of the effect of the Gaussian approximation in equations (5.26) and (5.27) on the integrals and the pseudo-likelihood evaluation is provided in Appendix A. For different kinds of random finite sets the accuracy of the approximation varies. The approximation is most accurate for Poisson/IIDC RFS, followed by union of multi-Bernoulli and IIDC RFS, and is least accurate for the multi-Bernoulli random finite sets.

5.1.3 Example: Multi-Bernoulli RFS

In this example we develop the approximate PHD update expression we obtained above when the predicted multitarget density at time $k+1$ corresponds to a multi-Bernoulli random finite set. Let $\Xi_{k+1|k}$ be a multi-Bernoulli random finite set with parameters $\{r_i, s_i(\mathbf{x})\}_{i=1}^{N_{k+1|k}}$. Thus from from (2.41) and (2.43) we have

$$D_{k+1|k}(\mathbf{x}) = \sum_{i=1}^{N_{k+1|k}} r_i \, s_i(\mathbf{x}) \tag{5.30}$$

$$\tilde{D}_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\}) = D_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\}) - D_{k+1|k}(\mathbf{x}_1) D_{k+1|k}(\mathbf{x}_2) \tag{5.31}$$

$$= - \sum_{i=1}^{N_{k+1|k}} r_i^2 \, s_i(\mathbf{x}_1) s_i(\mathbf{x}_2) \tag{5.32}$$

Combining the above expressions and Campbell's theorem, we get

$$\mathbf{m}_{k+1} = \int_{\mathcal{X}} g(\mathbf{x}) \, D_{k+1|k}(\mathbf{x}) d\mathbf{x} \tag{5.33}$$

$$= \int_{\mathcal{X}} g(\mathbf{x}) \left(\sum_{i=1}^{N_{k+1|k}} r_i \, s_i(\mathbf{x}) \right) d\mathbf{x} \tag{5.34}$$

$$= \sum_{i=1}^{N_{k+1|k}} r_i \int_{\mathcal{X}} g(\mathbf{x}) \, s_i(\mathbf{x}) d\mathbf{x} \tag{5.35}$$

$$= \sum_{i=1}^{N_{k+1|k}} r_i \, \mathbf{q}_i \,, \tag{5.36}$$

where $\mathbf{q}_i = \langle s_i, g \rangle$, and

$$\Sigma_{k+1} = \int_{\mathcal{X}} g(\mathbf{x}) g(\mathbf{x})^T D_{k+1|k}(\mathbf{x}) d\mathbf{x} + \int_{\mathcal{X}} \int_{\mathcal{X}} g(\mathbf{x}_1) g(\mathbf{x}_2)^T \tilde{D}_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\}) d\mathbf{x}_1 d\mathbf{x}_2 \tag{5.37}$$

$$= \int_{\mathcal{X}} g(\mathbf{x}) g(\mathbf{x})^T \left(\sum_{i=1}^{N_{k+1|k}} r_i \, s_i(\mathbf{x}) \right) d\mathbf{x} - \int_{\mathcal{X}} \int_{\mathcal{X}} g(\mathbf{x}_1) g(\mathbf{x}_2)^T \left(\sum_{i=1}^{N_{k+1|k}} r_i^2 \, s_i(\mathbf{x}_1) s_i(\mathbf{x}_2) \right) d\mathbf{x}_1 d\mathbf{x}_2 \tag{5.38}$$

$$= \sum_{i=1}^{N_{k+1|k}} r_i \int_{\mathcal{X}} s_i(\mathbf{x}) g(\mathbf{x}) g(\mathbf{x})^T d\mathbf{x} - \sum_{i=1}^{N_{k+1|k}} r_i^2 \left(\int_{\mathcal{X}} s_i(\mathbf{x}_1) g(\mathbf{x}_1) d\mathbf{x}_1 \right) \left(\int_{\mathcal{X}} s_i(\mathbf{x}_2) g(\mathbf{x}_2)^T d\mathbf{x}_2 \right) \tag{5.39}$$

$$= \sum_{i=1}^{N_{k+1|k}} \left(r_i \, v_i - r_i^2 \, \mathbf{q}_i \mathbf{q}_i^T \right), \tag{5.40}$$

where $v_i = \langle s_i, g g^T \rangle$.

We now express the multi-Bernoulli RFS $\Xi_{k+1|k}$ as $\Xi_{k+1|k} = \Xi^A_{k+1|k} \cup \Xi^B_{k+1|k}$ where $\Xi^A_{k+1|k}$ is the Bernoulli RFS with parameters $\{r_i, s_i(\mathbf{x})\}$ and $\Xi^B_{k+1|k}$ is the multi-Bernoulli RFS with the remaining parameter set $\{r_j, s_j(\mathbf{x})\}_{j \neq i}$. The random finite sets $\Xi^A_{k+1|k}$ and $\Xi^B_{k+1|k}$ are mutually independent. From (5.10) and Section 2.3.3, the density function $f^{A_{\mathbf{x}}}_{k+1|k}(W)$ corresponds to an RFS which is empty with probability one and all its moments are zero. Hence the parameters $\mathbf{m}^{A^*_{\mathbf{x}}}_{k+1}$ and $\Sigma^{A^*_{\mathbf{x}}}_{k+1}$, using Campbell's theorem, are

$$\mathbf{m}^{A^*_{\mathbf{x}}}_{k+1} = 0 + \sum_{j=1, j \neq i}^{N_{k+1|k}} r_j \, \mathbf{q}_j \tag{5.41}$$

$$= \mathbf{m}_{k+1} - r_i \, \mathbf{q}_i \tag{5.42}$$

$$\stackrel{\text{def}}{=} \mathbf{m}^{\bar{i}}_{k+1} \tag{5.43}$$

$$\Sigma^{A^*_{\mathbf{x}}}_{k+1} = 0 + \sum_{j=1, j \neq i}^{N_{k+1|k}} \left(r_j \, v_j - r_j^2 \, \mathbf{q}_j \mathbf{q}_j^T \right) \tag{5.44}$$

$$= \Sigma_{k+1} - \left(r_i \, v_i - r_i^2 \, \mathbf{q}_i \mathbf{q}_i^T \right) \tag{5.45}$$

$$\stackrel{\text{def}}{=} \Sigma^{\bar{i}}_{k+1} \tag{5.46}$$

From these equations we see that the quantities $\mathbf{m}^{A^*_{\mathbf{x}}}_{k+1}$ and $\Sigma^{A^*_{\mathbf{x}}}_{k+1}$ do not depend on $\mathbf{x}$ when

$\Xi^{A}_{k+1|k}$ is a Bernoulli RFS.

5.1.4 Example: Union of Multi-Bernoulli RFS and IIDC RFS

Let the RFS $\Xi_{k+1|k}$ be the union of a multi-Bernoulli RFS with parameters $\{r_i, s_i(\mathbf{x})\}_{i=1}^{N_{k+1|k}}$ and an IIDC RFS with parameters $\{s_c(\mathbf{x}), \pi^c(n)\}$ which are independent random finite sets. Let μ_c be the mean cardinality of IIDC RFS. Using the expressions for PHD and second factorial moments of IIDC and multi-Bernoulli random finite sets from Sections 2.3.2 and 2.3.4 and the results in equations (2.24) and (2.25) we have

$$D_{k+1|k}(\mathbf{x}) = \sum_{i=1}^{N_{k+1|k}} r_i\, s_i(\mathbf{x}) + \mu_c\, s_c(\mathbf{x}) \tag{5.47}$$

$$\tilde{D}_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\}) = D_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\}) - D_{k+1|k}(\mathbf{x}_1) D_{k+1|k}(\mathbf{x}_2) \tag{5.48}$$

$$= \left(a - \mu_c^2\right) s_c(\mathbf{x}_1) s_c(\mathbf{x}_2) - \sum_{i=1}^{N_{k+1|k}} r_i^2\, s_i(\mathbf{x}_1) s_i(\mathbf{x}_2) \tag{5.49}$$

where a is defined from equation (2.35) as

$$a \stackrel{\text{def}}{=} \sum_{n \geq 0} n(n-1)\, \pi^c(n). \tag{5.50}$$

Combining the above expressions and Campbell's theorem, we get

$$\mathbf{m}_{k+1} = \int_{\mathcal{X}} g(\mathbf{x}) \left(\sum_{i=1}^{N_{k+1|k}} r_i\, s_i(\mathbf{x}) + \mu_c\, s_c(\mathbf{x}) \right) d\mathbf{x} \tag{5.51}$$

$$= \sum_{i=1}^{N_{k+1|k}} r_i \int_{\mathcal{X}} s_i(\mathbf{x}) g(\mathbf{x}) d\mathbf{x} + \mu_c \int_{\mathcal{X}} s_c(\mathbf{x}) g(\mathbf{x}) d\mathbf{x} \tag{5.52}$$

$$= \sum_{i=1}^{N_{k+1|k}} r_i\, \mathbf{q}_i + \mu_c\, \mathbf{q}_c \tag{5.53}$$

where $\mathbf{q}_i = \langle s_i, g \rangle$ and $\mathbf{q}_c = \langle s_c, g \rangle$, and

$$\Sigma_{k+1} = \int_{\mathcal{X}} g(\mathbf{x})g(\mathbf{x})^T \left(\sum_{i=1}^{N_{k+1|k}} r_i \, s_i(\mathbf{x}) + \mu_c \, s_c(\mathbf{x}) \right) d\mathbf{x}$$
$$- \int_{\mathcal{X}} \int_{\mathcal{X}} g(\mathbf{x}_1)g(\mathbf{x}_2)^T \left(\sum_{i=1}^{N_{k+1|k}} r_i^2 \, s_i(\mathbf{x}_1)s_i(\mathbf{x}_2) \right) d\mathbf{x}_1 d\mathbf{x}_2$$
$$+ \int_{\mathcal{X}} \int_{\mathcal{X}} g(\mathbf{x}_1)g(\mathbf{x}_2)^T \left(a - \mu_c^2 \right) s_c(\mathbf{x}_1)s_c(\mathbf{x}_2) \, d\mathbf{x}_1 d\mathbf{x}_2 \tag{5.54}$$
$$= \sum_{i=1}^{N_{k+1|k}} (r_i \, v_i - r_i^2 \, \mathbf{q}_i \mathbf{q}_i^T) + \mu_c v_c - \left(\mu_c^2 - a \right) \mathbf{q}_c \mathbf{q}_c^T \tag{5.55}$$

where $v_i = \langle s_i, gg^T \rangle$ and $v_c = \langle s_c, gg^T \rangle$.

Now, express the random finite set $\Xi_{k+1|k}$ as the union $\Xi_{k+1|k} = \Xi^A_{k+1|k} \cup \Xi^B_{k+1|k} \cup \Xi^C_{k+1|k}$ where $\Xi^A_{k+1|k}$ is the Bernoulli RFS with parameters $\{r_i, s_i(\mathbf{x})\}$, $\Xi^B_{k+1|k}$ is the multi-Bernoulli RFS with the parameter set $\{r_j, s_j(\mathbf{x})\}_{j \neq i}$, and $\Xi^C_{k+1|k}$ is the IIDC RFS with parameters $\{s_c(\mathbf{x}), \pi^c(n)\}$. The random finite sets $\Xi^A_{k+1|k}$, $\Xi^B_{k+1|k}$, and $\Xi^C_{k+1|k}$ are mutually independent. Reasoning as before, the density function $f^{A_{\mathbf{x}}}_{k+1|k}(W)$ corresponds to an RFS which is empty with probability one and all its moments are zero. Hence the parameters $\mathbf{m}^{A^*_{\mathbf{x}}}_{k+1}$ and $\Sigma^{A^*_{\mathbf{x}}}_{k+1}$ using Campbell's theorem are

$$\mathbf{m}^{A^*_{\mathbf{x}}}_{k+1} = 0 + \sum_{j=1, j \neq i}^{N_{k+1|k}} r_j \, \mathbf{q}_j + \mu_c \, \mathbf{q}_c \tag{5.56}$$
$$= \mathbf{m}_{k+1} - r_i \, \mathbf{q}_i \tag{5.57}$$
$$\stackrel{\text{def}}{=} \mathbf{m}^{\bar{i}}_{k+1}, \tag{5.58}$$

and

$$\Sigma^{A^*_{\mathbf{x}}}_{k+1} = 0 + \sum_{j=1, j \neq i}^{N_{k+1|k}} (r_j \, v_j - r_j^2 \, \mathbf{q}_j \mathbf{q}_j T) + \mu_c v_c - \left(\mu_c^2 - a \right) \mathbf{q}_c \mathbf{q}_c^T \tag{5.59}$$
$$= \Sigma_{k+1} - (r_i \, v_i - r_i^2 \, \mathbf{q}_i \mathbf{q}_i^T) \tag{5.60}$$
$$\stackrel{\text{def}}{=} \Sigma^{\bar{i}}_{k+1}. \tag{5.61}$$

Alternatively we can express the random finite set $\Xi_{k+1|k}$ as $\Xi_{k+1|k} = \Xi^A_{k+1|k} \cup \Xi^B_{k+1|k}$ where $\Xi^A_{k+1|k}$ is the IIDC RFS with parameters $\{s_c(\mathbf{x}), \pi^c(n)\}$ and $\Xi^B_{k+1|k}$ is the multi-Bernoulli

RFS with parameter set $\{r_j, s_j(\mathbf{x})\}_{j=1}^{N_{k+1|k}}$. In this case the multitarget density $f_{k+1|k}^{A_{\mathbf{x}}}(W)$ corresponds to the IIDC RFS with parameters $\{s_c(\mathbf{x}), \frac{(n+1)\pi^c(n+1)}{\mu_c}\}$ with probability one. This is because, when $\mathbf{x} \notin W$ (event with probability one), using the expressions for density and PHD of an IIDC RFS from Section 2.3.2 we have

$$f_{k+1|k}^{A_{\mathbf{x}}}(W) = \frac{f_{k+1|k}^{A}(\{\mathbf{x}\} \cup W)}{D_{k+1|k}^{A}(\mathbf{x})} \tag{5.62}$$

$$= \frac{(|W|+1)!\,\pi^c(|W|+1)\,s_c^{\{\mathbf{x}\}\cup W}}{\mu_c\, s_c(\mathbf{x})} \tag{5.63}$$

$$= \frac{(|W|+1)!\,\pi^c(|W|+1)\,s_c^{W}}{\mu_c} \tag{5.64}$$

$$= |W|!\frac{(|W|+1)\pi^c(|W|+1)}{\mu_c}\, s_c^{W} \tag{5.65}$$

$$= |W|!\,\tilde{\pi}^c(|W|)\,s_c^{W}\,, \tag{5.66}$$

where $\tilde{\pi}^c(|W|) = \frac{(|W|+1)\,\pi^c(|W|+1)}{\mu_c}$ is a valid cardinality distribution as it sums to one. For the cardinality distribution $\tilde{\pi}^c(n)$ we have

$$\tilde{\mu}_c = \sum_{n=0}^{\infty} n\,\tilde{\pi}^c(n) \tag{5.67}$$

$$= \sum_{n=0}^{\infty} n\,\frac{(n+1)\,\pi^c(n+1)}{\mu_c} \tag{5.68}$$

$$= \frac{1}{\mu_c}\sum_{n=1}^{\infty} n\,(n-1)\,\pi^c(n) = \frac{a}{\mu_c} \tag{5.69}$$

where a is defined in (2.35), and

$$\tilde{a} = \sum_{n=2}^{\infty} n\,(n-1)\,\tilde{\pi}^c(n) \tag{5.70}$$

$$= \sum_{n=2}^{\infty} n\,(n-1)\,\frac{(n+1)\,\pi^c(n+1)}{\mu_c} \tag{5.71}$$

$$= \frac{1}{\mu_c}\sum_{n=3}^{\infty} n\,(n-1)\,(n-2)\,\pi^c(n) = \frac{b}{\mu_c}, \tag{5.72}$$

where

$$b = \sum_{n=3}^{\infty} n\,(n-1)\,(n-2)\,\pi^{c}(n). \tag{5.73}$$

Thus applying the results in (2.24) and (2.25) for the union of independent random finite sets and using Campbell's theorem

$$\mathbf{m}_{k+1}^{A_{\mathbf{x}}^{*}} = \sum_{j=1}^{N_{k+1|k}} r_j\,\mathbf{q}_j + \tilde{\mu}_c\,\mathbf{q}_c \tag{5.74}$$

$$= \sum_{j=1}^{N_{k+1|k}} r_j\,\mathbf{q}_j + \frac{a}{\mu_c}\,\mathbf{q}_c \tag{5.75}$$

$$\stackrel{\text{def}}{=} \mathbf{m}_{k+1}^{\bar{c}}, \tag{5.76}$$

and

$$\Sigma_{k+1}^{A_{\mathbf{x}}^{*}} = \sum_{j=1}^{N_{k+1|k}} \left(r_j\,v_j - r_j^2\,\mathbf{q}_j\mathbf{q}_j^{T}\right) + \tilde{\mu}_c v_c - \left(\tilde{\mu}_c^2 - \tilde{a}\right)\mathbf{q}_c\mathbf{q}_c^{T} \tag{5.77}$$

$$= \sum_{j=1}^{N_{k+1|k}} \left(r_j\,v_j - r_j^2\,\mathbf{q}_j\mathbf{q}_j^{T}\right) + \frac{a}{\mu_c}v_c - \left(\frac{a^2}{\mu_c^2} - \frac{b}{\mu_c}\right)\mathbf{q}_c\mathbf{q}_c^{T} \tag{5.78}$$

$$\stackrel{\text{def}}{=} \Sigma_{k+1}^{\bar{c}}. \tag{5.79}$$

5.2 Multi-Bernoulli filter

The multi-Bernoulli filter models the multitarget state as the union of multiple independent Bernoulli random finite sets. The scalar existence probability and the single target state density for each Bernoulli component are propagated over time. The propagation is done in two stages, prediction and update. The model for target dynamics accounts for the survival of existing targets from the previous time step to the current time step and for the birth of new targets. The single target motion model is used for propagation of surviving targets in the prediction step. Target birth is modeled as a multi-Bernoulli RFS. The most recent observation along with the superpositional sensor likelihood model is used in the update step to propagate the Bernoulli parameters.

5.2.1 Prediction step

The multi-Bernoulli prediction equations are derived in [34, 142]. Since the superpositional observation model does not play a role in the prediction step, the multi-Bernoulli prediction equations remain the same. We briefly review these equations in this section.

Let the posterior existence probability and state density parameters of the $N_{k|k}$ targets at time k be $\{r_{k|k,i}, s_{k|k,i}(\mathbf{x})\}_{i=1}^{N_{k|k}}$. At time $k+1$ let there be $N_{k+1|k}$ predicted targets with parameters $r_i \equiv r_{k+1|k,i}$ and $s_i(\mathbf{x}) \equiv s_{k+1|k,i}(\mathbf{x})$. Additionally, the predicted multi-Bernoulli RFS parameters can be expressed as

$$\{r_i, s_i(\mathbf{x})\}_{i=1}^{N_{k+1|k}} = \{r_i^P, s_i^P(\mathbf{x})\}_{i=1}^{N_{k|k}} \cup \{r_i^B, s_i^B(\mathbf{x})\}_{i=N_{k|k}+1}^{N_{k+1|k}} \tag{5.80}$$

where $\{r_i^P, s_i^P(\mathbf{x})\}_{i=1}^{N_{k|k}}$ are the parameters of targets propagated from the previous time step and $\{r_i^B, s_i^B(\mathbf{x})\}_{i=N_{k|k}+1}^{N_{k+1|k}}$ are the parameters of newly born targets. The relation between the the predicted target parameters at time $k+1$ and the posterior target parameters at time k is

$$r_i^P = r_{k|k,i} \int_{\mathcal{X}} s_{k|k,i}(\mathbf{x})\, p_{sv,k+1}(\mathbf{x}) d\mathbf{x} \tag{5.81}$$

$$= r_{k|k,i} \left\langle s_{k|k,i}, p_{sv,k+1} \right\rangle \tag{5.82}$$

$$s_i^P(\mathbf{x}) = \frac{\int_{\mathcal{X}} t_{k+1|k}(\mathbf{x}|\mathbf{w})\, s_{k|k,i}(\mathbf{w}) p_{sv,k+1}(\mathbf{w}) d\mathbf{w}}{\left\langle s_{k,i}, p_{sv,k+1} \right\rangle}, \tag{5.83}$$

where $p_{sv,k+1}(\mathbf{x})$ is the target survival probability and $t_{k+1|k}(\mathbf{x}|\mathbf{w})$ is the Markov transition kernel. Since the parameters $\{r_i^B, s_i^B(\mathbf{x})\}_{i=N_{k|k}+1}^{N_{k+1|k}}$ are used to model the new targets arriving at time $k+1$, they are initialized using the target birth model.

5.2.2 Update step

We assume that the posterior multi-target density also has the multi-Bernoulli form. For the case of superpositional sensors, the measurements do not provide any direct information about the number of targets. Hence no new Bernoulli components are added in the update step and $N_{k+1|k+1} = N_{k+1|k}$. Since the collective PHD of all the Bernoulli components can completely specify the posterior multi-Bernoulli density, the update step consists of updating the PHD for each of the $i = 1, 2, \ldots, N_{k+1|k+1}$ Bernoulli components. Let $\{r_i', s_i'(\mathbf{x})\}_{i=1}^{N_{k+1|k+1}}$

denote the parameter set of the posterior multi-Bernoulli density at time $k+1$. Combining the parameters derived in Section 5.1.3 and the approximate PHD update expression from (5.29), for the i^{th} Bernoulli component

$$r'_i\, s'_i(\mathbf{x}) \approx r_i\, s_i(\mathbf{x}) \frac{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma^{\bar{i}}_{k+1}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \mathbf{m}^{\bar{i}}_{k+1})}{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma_{k+1}}(\mathbf{z}_{k+1} - \mathbf{m}_{k+1})} \tag{5.84}$$

where,

$$\mathbf{m}_{k+1} = \sum_{i=1}^{N_{k+1|k}} r_i\, \mathbf{q}_i \tag{5.85}$$

$$\Sigma_{k+1} = \sum_{i=1}^{N_{k+1|k}} \left(r_i\, v_i - r_i^2\, \mathbf{q}_i \mathbf{q}_i^T\right) \tag{5.86}$$

$$\mathbf{m}^{\bar{i}}_{k+1} = \mathbf{m}_{k+1} - r_i\, \mathbf{q}_i \tag{5.87}$$

$$\Sigma^{\bar{i}}_{k+1} = \Sigma_{k+1} - \left(r_i\, v_i - r_i^2\, \mathbf{q}_i \mathbf{q}_i^T\right) \tag{5.88}$$

$$\mathbf{q}_i = \langle s_i, g \rangle, \quad v_i = \langle s_i, g g^T \rangle. \tag{5.89}$$

5.3 Hybrid multi-Bernoulli CPHD filter

The multi-Bernoulli RFS modeling of the multitarget state allows us to model each of the targets individually and update its state information. Although this can be seen as an improvement over the IIDC RFS modeling of the multitarget state which utilizes only one state density function to model all of the targets, it does not have a robust cardinality representation. Also, since the number of targets is changing over time, we need to add multiple Bernoulli components at each time step to account for target births. Processing a large number of Bernoulli components at each time step is not computationally efficient. To address these drawbacks we propose to use a hybrid approach where the existing targets are modeled using a multi-Bernoulli RFS and the newborn targets are modeled using the IIDC RFS.

The hybrid multi-Bernoulli CPHD filter uses the following modeling scheme. The final posterior distribution from the previous time step is modeled as a multi-Bernoulli RFS. In the prediction step, the multi-Bernoulli component is propagated following the motion model of surviving targets whereas to account for newborn targets an IIDC RFS compo-

nent is initialized. The IIDC component is independent of the multi-Bernoulli component. Thus the predicted distribution corresponds to the union of an IIDC random finite set and a multi-Bernoulli random finite set and these sets are independent. The union is completely represented by the PHD of the Bernoulli components, the PHD of the IIDC component and the cardinality distribution of the IIDC component. The update step propagates all of these quantities using Bayes' rule. Hence the obtained posterior is the union of an IIDC component and a multi-Bernoulli component. Since individual targets are better represented using Bernoulli random finite sets, the updated IIDC component is then approximated using multiple Bernoulli components. Thus the final posterior distribution is modeled using a multi-Bernoulli random finite set.

5.3.1 Prediction step

Let the parameters of the posterior Bernoulli components at time step k be denoted $\{r_{k|k,i}, s_{k|k,i}(\mathbf{x})\}_{i=1}^{N_{k|k}}$ as before. No new Bernoulli components are added in the prediction step to account for the birth of new targets, hence $N_{k+1|k} = N_{k|k}$. The Bernoulli parameters at the end of the prediction step are $\{r_i, s_i(\mathbf{x})\}_{i=1}^{N_{k+1|k}}$ and are given by equations (5.82) and (5.83). Let $\pi_{k+1|k}^{c}(n)$ and $s_c(\mathbf{x}) \equiv s_{k+1|k,c}(\mathbf{x})$ be the predicted cardinality distribution and the predicted density function at time $k+1$. Their exact forms depend on the specific target birth model used. Let μ_c denote the expected cardinality of the predicted IIDC RFS component.

5.3.2 Update step

The update step consists of updating the PHD for each of the Bernoulli components, the PHD of the IIDC component and the cardinality distribution of the IIDC component. Let the parameters of the posterior multi-Bernoulli and IIDC random finite sets be denoted by $\{r'_i, s'_i(\mathbf{x})\}_{i=1}^{N_{k+1|k+1}}$ and $\{s'_c(\mathbf{x}), \pi_{k+1}^{c}(n)\}$ respectively.

Combining the parameters derived in Section 5.1.4 and the approximate PHD update expression from (5.29), the PHD update of the i^{th} Bernoulli component is given by

$$r'_i\, s'_i(\mathbf{x}) \approx r_i\, s_i(\mathbf{x}) \frac{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma_{k+1}^{\bar{i}}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \mathbf{m}_{k+1}^{\bar{i}})}{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma_{k+1}}(\mathbf{z}_{k+1} - \mathbf{m}_{k+1})} \tag{5.90}$$

where,

$$\mathbf{m}_{k+1} = \sum_{i=1}^{N_{k+1|k}} r_i \, \mathbf{q}_i + \mu_c \, \mathbf{q}_c \tag{5.91}$$

$$\Sigma_{k+1} = \sum_{i=1}^{N_{k+1|k}} \left(r_i \, v_i - r_i^2 \, \mathbf{q}_i \mathbf{q}_i^T \right) + \mu_c \, v_c - \left(\mu_c^2 - a \right) \mathbf{q}_c \mathbf{q}_c^T \tag{5.92}$$

$$\mathbf{m}_{k+1}^{\bar{i}} = \mathbf{m}_{k+1} - r_i \, \mathbf{q}_i \tag{5.93}$$

$$\Sigma_{k+1}^{\bar{i}} = \Sigma_{k+1} - \left(r_i \, v_i - r_i^2 \, \mathbf{q}_i \mathbf{q}_i^T \right) \tag{5.94}$$

$$\mathbf{q}_i = \langle s_i, g \rangle, \; v_i = \langle s_i, g g^T \rangle \tag{5.95}$$

$$\mathbf{q}_c = \langle s_c, g \rangle, \; v_c = \langle s_c, g g^T \rangle \tag{5.96}$$

$$a = \sum_{n=2}^{\infty} n(n-1) \, \pi_{k+1|k}^{c}(n). \tag{5.97}$$

Similarly, combining the parameters derived in Section 5.1.4 and the approximate PHD update expression from (5.29), the PHD update for the IIDC RFS component is

$$\mu_c' \, s_c'(\mathbf{x}) \approx \mu_c \, s_c(\mathbf{x}) \frac{\mathcal{N}_{\Sigma_{\mathbf{z}} + \Sigma_{k+1}^{\bar{c}}}(\mathbf{z}_{k+1} - g(\mathbf{x}) - \mathbf{m}_{k+1}^{\bar{c}})}{\mathcal{N}_{\Sigma_{\mathbf{z}} + \Sigma_{k+1}}(\mathbf{z}_{k+1} - \mathbf{m}_{k+1})} \tag{5.98}$$

where,

$$\mathbf{m}_{k+1}^{\bar{c}} = \sum_{j=1}^{N_{k+1|k}} r_j \, \mathbf{q}_j + \frac{a}{\mu_c} \mathbf{q}_c \tag{5.99}$$

$$\Sigma_{k+1}^{\bar{c}} = \sum_{j=1}^{N_{k+1|k}} \left(r_j \, v_j - r_j^2 \, \mathbf{q}_j \mathbf{q}_j^T \right) + \frac{a}{\mu_c} v_c - \left(\frac{a^2}{\mu_c^2} - \frac{b}{\mu_c} \right) \mathbf{q}_c \mathbf{q}_c^T \tag{5.100}$$

$$b = \sum_{n=0}^{\infty} n(n-1)(n-2) \, \pi_{k+1|k}^{c}(n). \tag{5.101}$$

The parameters $\mathbf{m}_{k+1}$ and Σ_{k+1} are as given in (5.91) and (5.92) respectively.

IIDC component cardinality update

The main advantage of the IIDC component in the hybrid filter is that we can make use of the accurate cardinality estimation of the CPHD filter. The cardinality distribution of

the posterior IID cluster component is defined as

$$\pi^{c}_{k+1|k+1}(n) = \int_{|W|^c=n} f_{k+1|k+1}(W)\delta W \tag{5.102}$$

$$= \frac{\int_{|W|^c=n} h_{\mathbf{z}_{k+1}}(W) f_{k+1|k}(W)\delta W}{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(W) f_{k+1|k}(W)\delta W} \tag{5.103}$$

$$= \pi^{c}_{k+1|k}(n)\frac{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(W) f^{c,n}_{k+1|k}(W)\delta W}{\int_{\mathcal{X}} h_{\mathbf{z}_{k+1}}(W) f_{k+1|k}(W)\delta W} \tag{5.104}$$

$$\text{where} \quad f^{c,n}_{k+1|k}(W) = \frac{1}{\pi^{c}_{k+1|k}(n)}\delta_{|W|^c}(n) f_{k+1|k}(W) \tag{5.105}$$

The multi-target density $f^{c,n}_{k+1|k}(W)$ corresponds to the union of a multi-Bernoulli RFS and the random finite set obtained by constraining the cardinality ($|W|^c = n$) of the IID cluster RFS. Applying the approximations as before we get

$$\pi^{c}_{k+1}(n) \approx \pi^{c}_{k+1|k}(n)\frac{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma^{c,n}_{k+1}}(\mathbf{z}_{k+1} - \mathbf{m}^{c,n}_{k+1})}{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma_{k+1}}(\mathbf{z}_{k+1} - \mathbf{m}_{k+1})} \tag{5.106}$$

where,

$$\mathbf{m}^{c,n}_{k+1} = \sum_{i=1}^{N_{k+1|k}} r_i\,\mathbf{q}_i + n\,q_c \tag{5.107}$$

$$\Sigma^{c,n}_{k+1} = \sum_{i=1}^{N_{k+1|k}} (r_i\,v_i - r_i^2\,\mathbf{q}_i\mathbf{q}_i^T) + n(v_c - \mathbf{q}_c\mathbf{q}_c^T) \tag{5.108}$$

The parameters $\mathbf{m}_{k+1}$ and Σ_{k+1} are as given in equations (5.91) and (5.92), respectively. Note that in the above update equation there is no assumption made about the cardinality of the multi-Bernoulli component. The multi-Bernoulli filter can be treated as a special case of the hybrid multi-Bernoulli CPHD filter. Indeed, we obtain the multi-Bernoulli filter update equations if we set the IIDC component to be the empty set in all of the equations above.

5.4 Auxiliary particle filter implementations

We implement the proposed filters using a Monte Carlo approach. Approximate update equations have been derived in this chapter but even they do not lead to a fully analytically

tractable filter. Hence we develop particle filter based implementations of the filters. The basic particle filter approach does not give a stable implementation because of the multiple approximations employed to derive the filter equations. We propose auxiliary particle filter implementations of the multi-Bernoulli filter and the hybrid multi-Bernoulli filter based on the auxiliary particle filter implementation of the PHD filter discussed in [105].

The normalized posterior PHD corresponding to each Bernoulli component $s_{k-1|k-1,i}(\mathbf{x})$, and the IIDC component $s_{k-1|k-1,c}(\mathbf{x})$ at time $k-1$ is approximated using a set of weighted particles as follows

$$s_{k-1|k-1,\theta}(\mathbf{x}) \approx \hat{s}_{k-1|k-1,\theta}(\mathbf{x}) = \sum_{j=1}^{N_p} w_{k-1,\theta}^{(j)}\, \delta(\mathbf{x} - \mathbf{x}_{k-1,\theta}^{(j)}) \tag{5.109}$$

$$\sum_{j=1}^{N_p} w_{k-1,\theta}^{(j)} = 1\,, \tag{5.110}$$

where $\theta = i, i = 1 \dots N_{k-1|k-1}$ for Bernoulli components and $\theta = c$ for the IIDC component. The probabilities of existence of the Bernoulli components are $\hat{r}_{k-1,i}, i = 1 \dots N_{k-1|k-1}$ and the cardinality distribution of the IIDC component is represented using a finite dimensional vector $\hat{\pi}_{k-1}^{c}(n)$ whose elements sum to one. For the hybrid multi-Bernoulli CPHD filter the quantities $\hat{s}_{k-1|k-1,c}(\mathbf{x})$ and $\hat{\pi}_{k-1}^{c}(n)$ are initialized using the birth process parameters. For the CPHD filter these quantities are obtained from the previous time step. Thus one particle filter is used for each Bernoulli component and one particle filter is used to approximate the IIDC component. The pseudo-code for the auxiliary particle filter implementation of the hybrid multi-Bernoulli CPHD filter are provided in Figures 5.1 and 5.2.

For each Bernoulli component, at time k, the auxiliary variables $\bar{\mathbf{x}}_{k-1,i}^{(j)}$ are sampled from a mixture of a re-weighted particle set $p_{k-1,i}(\mathbf{x})$ and the posterior from the previous time step $\hat{s}_{k-1|k-1,i}(\mathbf{x})$. Similarly for the IIDC component the auxiliary variables $\bar{\mathbf{x}}_{k-1,c}^{(j)}$ are sampled from a mixture of a re-weighted particle set $p_{k-1,c}(\mathbf{x})$ and the posterior from the previous time step $\hat{s}_{k-1|k-1,c}(\mathbf{x})$. The re-weighted particle set $p_{k-1,\theta}(\mathbf{x})$ is given by

$$p_{k-1,\theta}(\mathbf{x}) = \sum_{j=1}^{N_p} \tilde{w}_{k-1,\theta}^{(j)}\, \delta(\mathbf{x} - \mathbf{x}_{k-1,\theta}^{(j)}), \tag{5.111}$$

$$\tilde{w}_{k-1,\theta}^{(j)} \propto w_{k-1,\theta}^{(j)} \left(\frac{\mathcal{N}_{\Sigma_{\mathbf{z}}+\tilde{\Sigma}_k^{\bar{\theta}}}(\mathbf{z}_k - g(\psi(\mathbf{x}_{k-1,\theta}^{(j)})) - \tilde{\mathbf{m}}_k^{\bar{\theta}})}{\mathcal{N}_{\Sigma_{\mathbf{z}}+\tilde{\Sigma}_k}(\mathbf{z}_k - \tilde{\mathbf{m}}_k)} \right)^{\epsilon}, \tag{5.112}$$

where $\sum_{j=1}^{N_p} \tilde{w}_{k-1,\theta}^{(j)} = 1$; $\psi(\mathbf{x}_{k-1,\theta}^{(j)}) = \mathbb{E}(\mathbf{x}_{k,\theta}|\mathbf{x}_{k-1,\theta}^{(j)})$; the quantities $\tilde{\mathbf{m}}_k$, $\tilde{\mathbf{m}}_k^{\bar{\theta}}$, $\tilde{\Sigma}_k$ and $\tilde{\Sigma}_k^{\bar{\theta}}$ are calculated using (5.91)-(5.97) and (5.99)-(5.101) with particle approximations for q_θ and v_θ evaluated using the particle set $\psi(\mathbf{x}_{k-1,\theta}^{(j)})$; and ϵ is the tempering factor [105] for stabilizing the weights in the auxiliary particle filter. Regularization is performed by adding a small zero-mean Gaussian jitter $\Delta_{k,\theta}^{(j)} \sim \mathcal{N}_{\Sigma_{\text{reg}}}(0)$ to the particles to maintain their diversity and avoid particle degeneracy.

for $k = 1$ to T **do**
  Sample auxiliary particles
  Construct $p_{k-1,\theta}(\mathbf{x})$ using (5.111), $\theta = i, c$
  for $j = 1$ to N_p **do**
    for $i = 1$ to $N_{k-1|k-1}$ **do**
      $\bar{\mathbf{x}}_{k-1,i}^{(j)} \sim \alpha\, p_{k-1,i}(\mathbf{x}) + (1-\alpha)\, \hat{s}_{k-1|k-1,i}(\mathbf{x})$
      Regularization: $\bar{\mathbf{x}}_{k-1,i}^{(j)} = \bar{\mathbf{x}}_{k-1,i}^{(j)} + \Delta_{k,i}^{(j)}$
    end for
    $\bar{\mathbf{x}}_{k-1,c}^{(j)} \sim \alpha\, p_{k-1,c}(\mathbf{x}) + (1-\alpha)\, \hat{s}_{k-1|k-1,c}(\mathbf{x})$
    Regularization: $\bar{\mathbf{x}}_{k-1,c}^{(j)} = \bar{\mathbf{x}}_{k-1,c}^{(j)} + \Delta_{k,c}^{(j)}$
  end for
  Proposal
  for $j = 1$ to N_p **do**
    for $i = 1$ to $N_{k-1|k-1}$ **do**
      $\mathbf{x}_{k,i}^{(j)} \sim t_{k|k-1}(\mathbf{x}|\bar{\mathbf{x}}_{k-1,i}^{(j)})$
    end for
    $\mathbf{x}_{k,c}^{(j)} \sim t_{k|k-1}(\mathbf{x}|\bar{\mathbf{x}}_{k-1,c}^{(j)})$
  end for
  $r_{k|k-1,i} = r_{k-1,i}\, p_s$, $i = 1, 2, \ldots, N_{k-1|k-1}$
  Update See Figure 5.2
  Approximation and track management See Figure 5.2
end for

Fig. 5.1 Pseudo-code for the auxiliary particle filter implementation of the hybrid multi-Bernoulli CPHD filter (auxiliary and proposal steps).

Update
for $j = 1$ to N_p **do**
  for $i = 1$ to $N_{k-1|k-1}$ **do**
    $w_{k,i}^{(j)} \propto w_{k-1,i}^{(j)} \frac{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma_k^{\bar{i}}}(\mathbf{z}_k - g(\mathbf{x}_{k,i}^{(j)}) - \mathbf{m}_k^{\bar{i}})}{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma_k}(\mathbf{z}_k - \mathbf{m}_k)}$
  end for
  $w_{k,c}^{(j)} \propto w_{k-1,c}^{(j)} \frac{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma_k^{\bar{c}}}(\mathbf{z}_k - g(\mathbf{x}_{k,c}^{(j)}) - \mathbf{m}_k^{\bar{c}})}{\mathcal{N}_{\Sigma_{\mathbf{z}}+\Sigma_k}(\mathbf{z}_k - \mathbf{m}_k)}$
end for
weight compensation $w_{k,\theta}^{(j)} \propto \frac{w_{k,\theta}^{(j)}}{\tilde{w}_{k-1,\theta}^{(j)}}$ for auxiliary particles sampled from re-weighted distribution
$\{\mathbf{x}_{k,i}^{(j)}, \frac{1}{N_p}\}$ = resample $(\{\mathbf{x}_{k,i}^{(j)}, w_{k,i}^{(j)}\}), i = 1 \dots N_{k-1|k-1}$
Update $r_{k,i}, i = 1 \dots N_{k-1|k-1}$ using (5.90)
Update $\hat{\pi}_k^c(n)$ using (5.106)
Approximation and track management
Prune Bernoulli tracks with $r_{k,i} < r_0, i = 1 \dots N_{k-1|k-1}$
$N_k^c = \text{MAP}\ (\hat{\pi}_k^c(n))$
$\{\mathbf{x}_{k,i}^{(j)}, \frac{1}{N_p}\}$ = split $(\{\mathbf{x}_{k,c}^{(j)}, w_{k,c}^{(j)}\}, N_{k|k}^c), r_{k,i} = 1,$
$i = N_{k-1|k-1} + 1 \dots N_{k-1|k-1} + N_{k|k}^c$
Gate new Bernoulli components to check for duplicity

Fig. 5.2 Pseudo-code for the auxiliary particle filter implementation of the hybrid multi-Bernoulli CPHD filter (update and approximation steps).

In the proposal (prediction) step the particles are propagated according to the target transition model $t_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$. For the update step the quantities $\mathbf{m}_k$, $\mathbf{m}_k^{\bar{\theta}}$, Σ_k and $\Sigma_k^{\bar{\theta}}$ are are calculated using (5.91)-(5.97) and (5.99)-(5.101). The PHD update step is realized by performing an update of the particle weights using equations (5.90) and (5.98). For the particles sampled from the re-weighted distribution $p_{k-1,\theta}(\mathbf{x})$, weight compensation is performed. For each Bernoulli component the weighted particle set is resampled to obtain particles with equal weights. The existence probability is updated from (5.90) by using a particle approximation for $s_i(\mathbf{x})$. The cardinality distribution of the IIDC component $\hat{\pi}_k^c(n)$ is updated using (5.106)-(5.108).

Pruning of the Bernoulli components is performed in order to eliminate targets with

low probability of existence ($< r_0$). Bernoulli components with existence probability lower than the existence probability at birth are pruned, i.e., $r_0 = r^B$. A low existence probability threshold is chosen because it can identify individual targets even when they are in close vicinity. An estimate of the number of newborn targets $N^c_{k|k}$ is obtained from the IIDC cardinality distribution $\hat{\pi}^c_k(n)$ using the maximum a posteriori (MAP) rule. The "split" function partitions the set of particles representing the normalized IIDC PHD into $N^c_{k|k}$ clusters using the k-means algorithm and each cluster is used to initialize a new Bernoulli component with existence probability 1. The new components created can sometimes correspond to spurious copies of existing targets. Hence gating is performed so that new targets starting within close vicinity of existing targets are pruned. The definition of "closeness" between two targets is application dependent. For example, in our radio frequency tomography application, we use the Euclidean distance measure between the centroids of the two particle sets representing the target positions and eliminate the new component if the distance is less than a meter. The pruned Bernoulli components are used for initialization of the IIDC PHD along with the target birth model in the next time step. This can be helpful in case of low detection probability or high noise variance.

The multi-Bernoulli auxiliary particle filter implementation is very similar to the pseudocode given in Figure 5.1. The major difference is that instead of the IIDC component, multiple Bernoulli components are initialized in the prediction stage to account for target births. In the update stages, the PHDs of the corresponding Bernoulli components are updated. There is no clustering step required but the pruning and the gating steps are the same. The CPHD auxiliary particle filter implementation is obtained by ignoring the steps related to the multi-Bernoulli component in Figure 5.1. The normalized posterior PHD is propagated to the next time step instead of approximating it with a multi-Bernoulli component.

The implementations of the multi-Bernoulli filters using particle filters are similar to that of the multiple particle filter (MPF) [159] in the sense that they all use one particle filter per target. The MPF assumes the number of targets to be fixed and known but the multi-Bernoulli filters automatically track the changing number of targets. The update step in the MPF propagates the marginal posterior whereas the update step in the multi-Bernoulli filters propagate the PHD of Bernoulli components.

5.5 Numerical simulations

In this section we use numerical simulations to demonstrate successful application of the proposed filters to the problem of multitarget tracking. We consider two examples of the superpositional sensor model. The first is a radio frequency tomography application and the second is an acoustic sensor network.

5.5.1 Target dynamics

We assume that for each target its dynamics are independent of the other targets and their dynamics. Specifically, motion of each target when present within the monitoring region is governed by the following approximately constant velocity model [36]:

$$\mathbf{x}_{k+1,i} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{k,i} + \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ T & 0 \\ 0 & T \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix} \tag{5.113}$$

where T is the sampling period and u_x, u_y are zero-mean Gaussian white noise with respective variance $\sigma^2_{u_x}$ and $\sigma^2_{u_y}$. In this model, the state of each object i at time k, $\mathbf{x}_{k,i}$, is represented by a four-dimensional vector: position on the x-axis and y-axis, velocity on the x-axis and y-axis. Multiple targets can be simultaneously present and targets can appear or disappear over time.

5.5.2 Algorithm settings

We compare the multi-Bernoulli (MBR) filter and the hybrid multi-Bernoulli CPHD (MBR-CPHD) filter with the CPHD filter. The auxiliary particle filter implementations of these filters are as discussed in Section 5.4. To compare the performance of different filters we use the OSPA error metric described in Section 2.7. The multitarget state estimate is obtained by averaging particles representing position for each existing Bernoulli component. For the CPHD filter the state estimates are centroids of the clusters obtained by partitioning the particle set using the k-means algorithm.

A single target survival probability is assumed to be constant throughout the monitoring region and is equal to $p_s = 0.9$. For the MBR filter four new Bernoulli components are

added at each time step to account for target births. The probability of existence of these new components are $r^B = 0.2$ and their density functions are uniform within the monitoring region. For the hybrid MBR-CPHD filter the birth process is IIDC with discrete uniform cardinality distribution and the normalized PHD is assumed uniform within the monitoring region. Rejected components from the current time step are also used to partially initialize the PHD of the IIDC component in the next time step. This way, targets with low probability of existence which get erroneously eliminated can be reintroduced using the IIDC component.

Gating is performed with a gating radius of r_g to check for duplicity between the new and existing targets. Bernoulli components with existence probability lower than the existence probability at birth are pruned, i.e., $r_0 = r^B = 0.2$. We use $N_p = 1000$ particles for each particle filter which is a standard value used in the literature, sufficient for single target tracking. In the auxiliary step we use $\alpha = 0.5$ which gives equal weights to the auxiliary and prior distributions. In the measurement models we study, the likelihood function can get peaky. Hence we use a low value of tempering factor, $\epsilon = 0.3$, to smooth the auxiliary weights. Regularization of particles is performed to keep the particle set sufficiently diverse with $\Sigma_{\text{reg}} = \sigma^2_{\text{reg}}\text{diag}(1, 1, 1, 1)$.

5.5.3 Radio frequency tomography

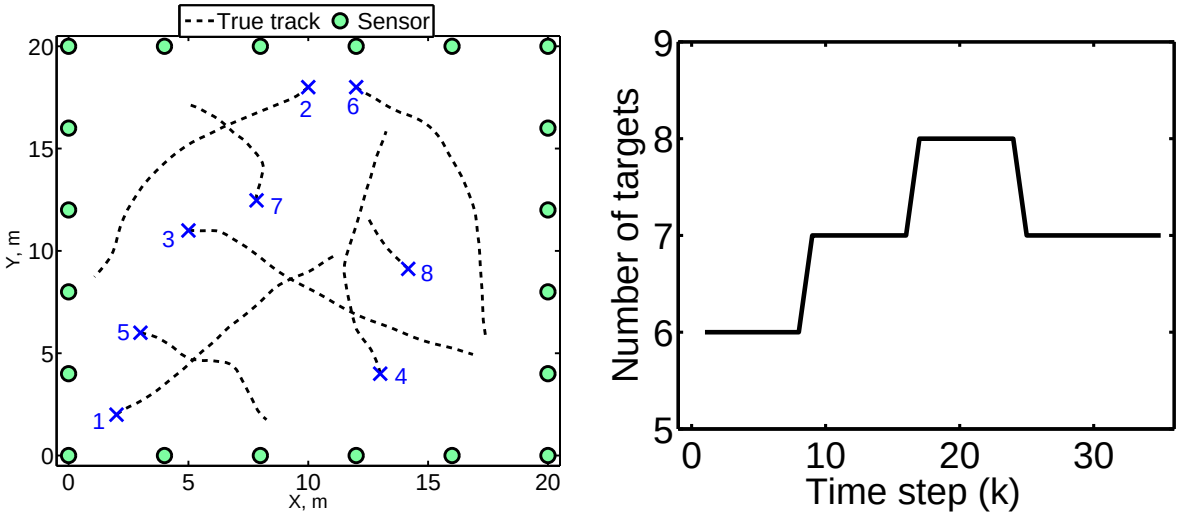
Measurement model

The radio frequency tomography sensor system is described in Section 3.1 and the measurement model we simulate here is discussed in Section 3.2.1. We simulate an RF sensor network with $N_s = 20$ sensor nodes distributed uniformly on the periphery of the $20m \times 20m$ square region as shown in Figure 5.3(a). This gives rise to a total of $n_{\mathbf{z}} = 190$ unique bidirectional links. The observation model parameters are $\phi = 5$ and $\sigma_\lambda = 0.4$. The measurement noise variance is $\Sigma_{\mathbf{z}} = \sigma^2_{\mathbf{z}} I_{n_{\mathbf{z}}}$ where $\sigma^2_{\mathbf{z}} = 0.25$ and $I_{n_{\mathbf{z}}}$ is the $n_{\mathbf{z}} \times n_{\mathbf{z}}$ identity matrix. These parameter values are based on the empirical values observed from experiments (Chapter 3).

Target tracks and filter parameters

Figure 5.3(a) shows the target tracks we use for the simulations. The black cross (x) indicates the starting location of the target. The variation of number of targets over time is shown in Figure 5.3(b). The targets labelled with the numbers 7 and 8 in Figure 5.6 appear

within the monitoring region at time steps 9 and 17 respectively. Target 8 disappears from the monitoring region at time step 24. The targets in the above scenario evolve according to the linear Gaussian dynamics given in Equation (5.113) with a time step of duration $T = 0.25s$ and the noise variance parameters of $\sigma^2_{u_x} = \sigma^2_{u_y} = 0.35$. We simulate 35 time steps of target motion for a total of $35 \times 0.25 = 8.75s$. The standard deviation of the regularization noise is set to $\sigma_{\text{reg}} = 0.25$ and the gating radius is set to $r_g = 1m$.



(a) Target tracks (b) Cardinality

Fig. 5.3 RF tomography: (a) Target tracks used in the simulations. (b) Variation of number of targets over time.

Simulation results

The average OSPA error metrics are calculated by repeating simulations multiple times with different random initializations. The target tracks shown in Figure 5.3(a) are used for all the Monte Carlo runs. A set of 20 different measurement sequences are generated and each is processed with 5 different random initializations for all the algorithms. Thus the average error is reported by running 100 Monte Carlo simulations. We ignore the first 5 time steps when calculating the average error to allow the filter estimates to stabilize. The mean OSPA error is calculated for different values of the measurement noise parameter $\sigma_{\mathbf{z}}$ and are shown in Figure 5.4 for $c = 0.5$ and $c = 1$. As the noise is increased from $\sigma_{\mathbf{z}} = 1$ to $\sigma_{\mathbf{z}} = 2$ the performance of all the filters deteriorates. When the measurement noise is

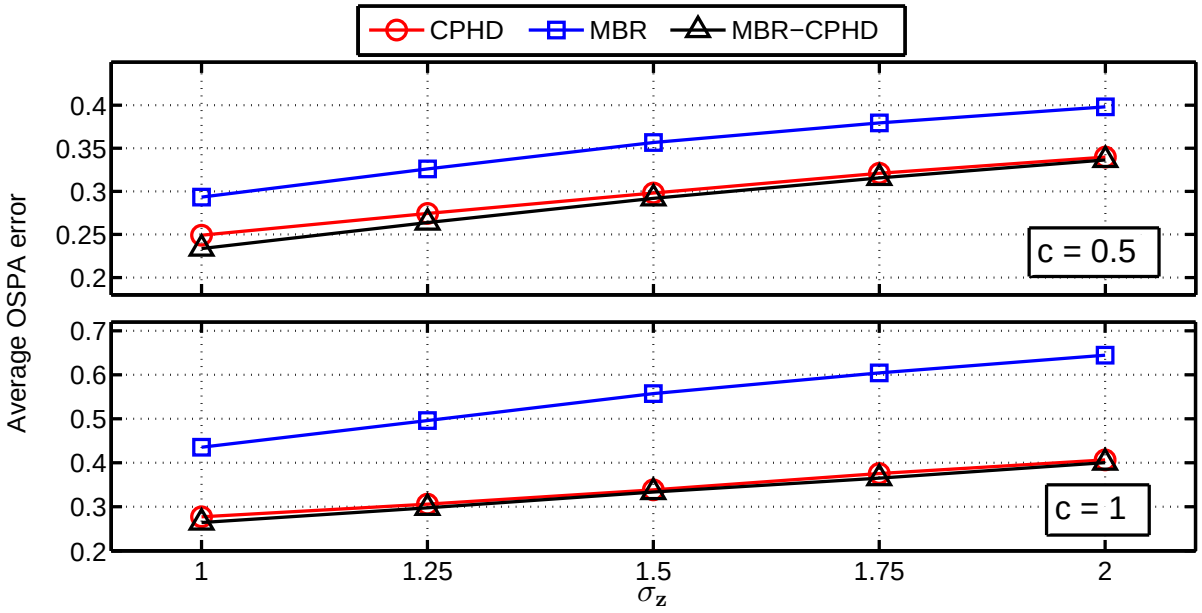


Fig. 5.4 RF tomography: The average OSPA error as the measurement noise standard deviation $\sigma_{\mathbf{z}}$ is increased from $\sigma_{\mathbf{z}} = 1$ to $\sigma_{\mathbf{z}} = 2$.

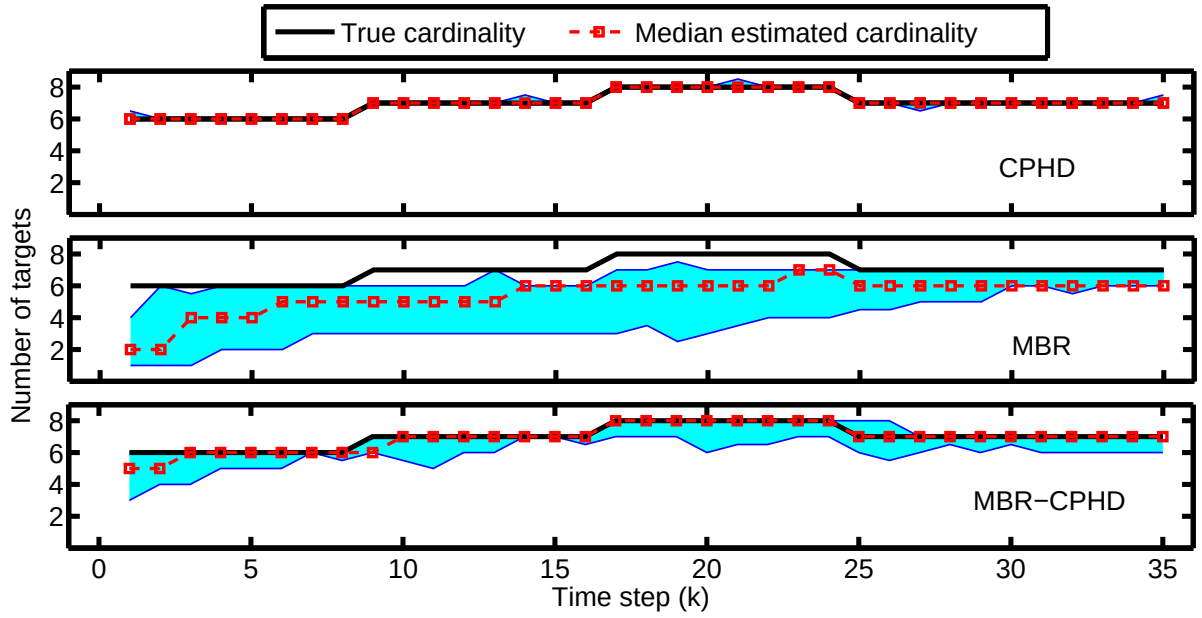


Fig. 5.5 RF tomography: Median cardinality and its 5-95 percentiles (shaded region) as a function of time for $\sigma_{\mathbf{z}} = 1.5$.

small the hybrid MBR-CPHD filter has the lowest error among all the filters. For higher measurement noise the CPHD filter and the hybrid MBR-CPHD filter have almost the same performance.

The median cardinality estimates (over the 100 Monte Carlo simulations) and its 5^{th} and 95^{th} percentiles for the different algorithms are shown in Figure 5.5 for $\sigma_{\mathbf{z}} = 1.5$. The multi-Bernoulli filter has low initial cardinality estimates because only a maximum of 4 new Bernoulli components are added at each time step. The MBR filter also significantly underestimates the number of targets which is reflected as high average OSPA error as seen from Figure 5.4. The CPHD filter has the most accurate cardinality estimate as its 5^{th} and 95^{th} percentiles coincide with the median cardinality at most of the time steps. The hybrid MBR-CPHD filter makes significantly better cardinality estimates than the MBR filter. The targets missed by the MBR-CPHD filter at the current time step are reintroduced by input from the CPHD component at the next time step.

Since the average OSPA error for the CPHD filter and the MBR-CPHD filter are almost the same and CPHD filter provides better cardinality estimates than the MBR-CPHD filter, the MBR-CPHD filter provides more accurate target location information than the CPHD filter. In fact, ignoring the errors in cardinality, the root-mean-square error averaged over 100 Monte Carlo simulations are $0.42m$, $0.54m$, and $0.32m$ for the CPHD, MBR and MBR-CPHD filters respectively when the measurement noise is $\sigma_{\mathbf{z}} = 1.5$. In this simulation the targets approach reasonably close to each other (within a distance of one meter). Since the approximate PHD update equation (5.21) is more accurate when the targets are well separated, this might explain the cardinality errors made by the hybrid MBR-CPHD filter. We expect these cardinality errors made by the MBR and hybrid MBR-CPHD filters to increase if the targets cross each other as the underlying assumption no longer holds and some of the components merge into a single component. Figure 5.6 depicts an example of target location estimates obtained using the hybrid MBR-CPHD filter.

5.5.4 Acoustic amplitude sensors

Measurement model

The acoustic sensor likelihood model is discussed in Section 2.4.2. We simulate an acoustic sensor network with $N_s = 25$ sensor nodes distributed in a $1000m \times 1000m$ square region in a grid format as shown in Figure 5.10. A wider observation region is chosen to evaluate

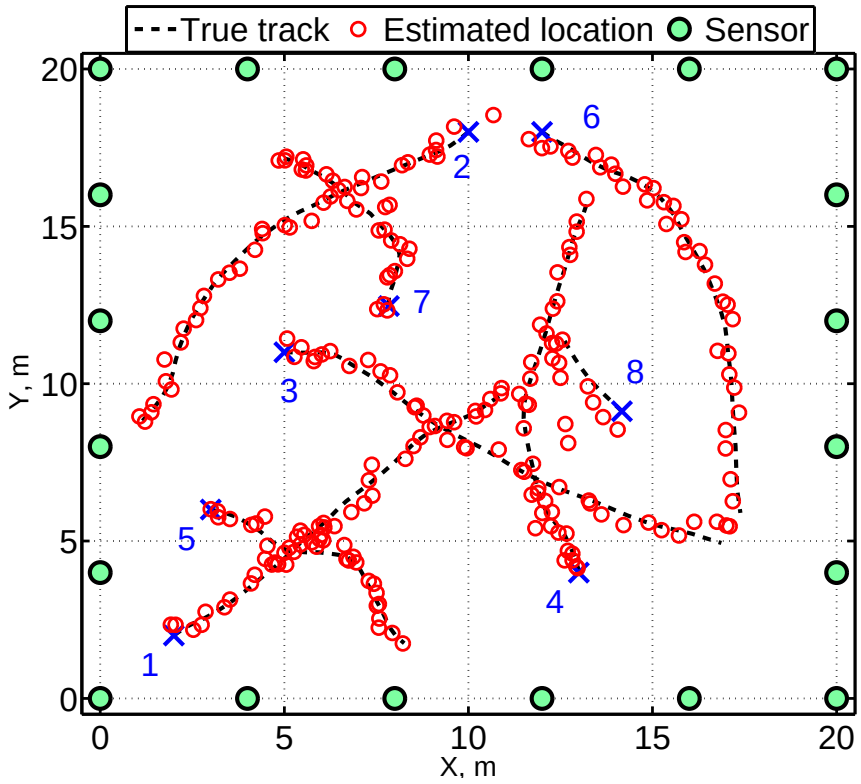


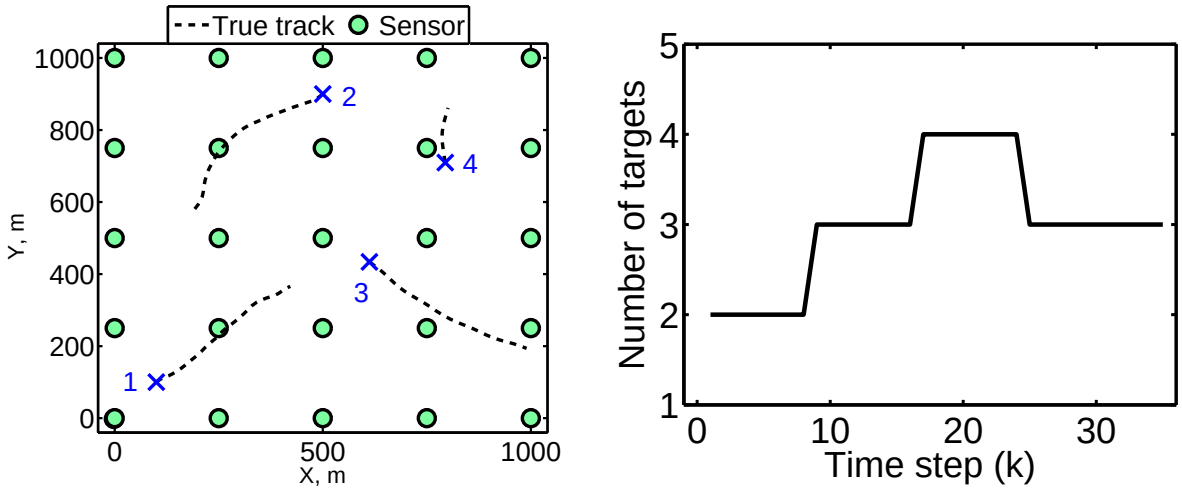
Fig. 5.6 RF tomography: True target tracks and estimated target locations obtained using the hybrid MBR-CPHD filter for $\sigma_{\mathbf{z}} = 1.5$.

the robustness of the filters in a more challenging scenario. The measurement dimension is $n_{\mathbf{z}} = N_s = 25$ and the measurement model parameters are $A = 500$, $\kappa = 1$ and $d_0 = 1$. The measurement noise variance is $\Sigma_{\mathbf{z}} = \sigma_{\mathbf{z}}^2 I_{n_{\mathbf{z}}}$.

Target tracks and filter parameters

The target dynamics discussed in Section 5.5.1 are used to simulate the target tracks. The simulated target tracks are shown in Figure 5.7(a) and the target number variation is shown in Figure 5.7(b). These tracks are simulated for 35 time steps using the process noise parameters of $\sigma_{u_x} = \sigma_{u_y} = 25m$ and time step duration $T = 0.25s$. The standard deviation of the regularization noise is set to $\sigma_{\text{reg}} = 5$. The gating radius is increased to $r_g = 100m$ since the monitored region is considerably larger. We note that the gating is used only for elimination of tracks spawning from existing targets and not for termination of existing target tracks. A high value of gating radius is used in this simulation because

the multi-Bernoulli filter gave rise to numerous spurious tracks for a smaller value of gating radius.



(a) Target tracks (b) Cardinality

Fig. 5.7 Acoustic sensor network: (a) Target tracks used in the simulations. (b) Variation of number of targets over time.

Simulation results

Figure 5.8 shows how the average OSPA error varies as a function of measurement noise for $c = 50$ and $c = 100$. The averages are calculated using 100 Monte Carlo simulations. The hybrid MBR-CPHD filter performs significantly better than the CPHD filter in this setup and has the lowest average OSPA error for all values of $\sigma_{\mathbf{z}}$. The median cardinality and the 5^{th} and 95^{th} percentiles are shown in Figure 5.9 for $\sigma_{\mathbf{z}} = 1$. The CPHD filter has the most accurate cardinality estimates followed by the hybrid MBR-CPHD filter and the MBR filter. Since the hybrid MBR-CPHD filter has lower average OSPA error than the CPHD filter, for the targets that are correctly identified, the hybrid MBR-CPHD filter is able to accurately track their locations in a much wider observation region. Figure 5.10 shows the true target trajectories and the estimated target locations obtained using the hybrid MBR-CPHD filter for $\sigma_{\mathbf{z}} = 1$.

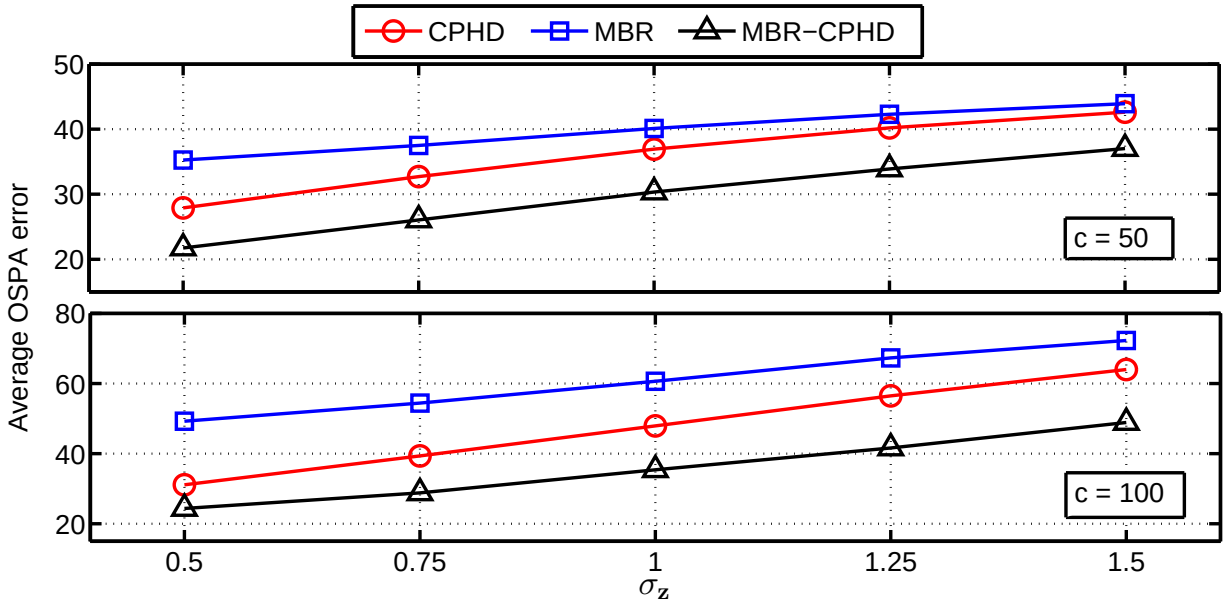


Fig. 5.8 Acoustic sensor network: The average OSPA error as the measurement noise standard deviation σ_z is increased from $\sigma_z = 0.5$ to $\sigma_z = 1.5$.

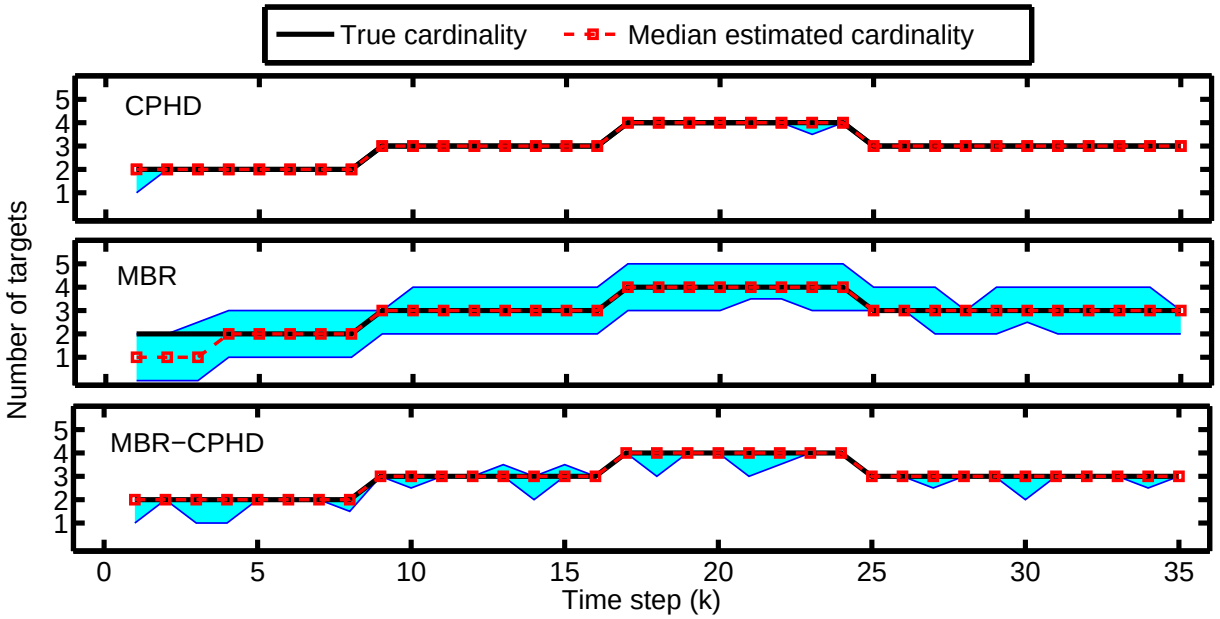


Fig. 5.9 Acoustic sensor network: Median cardinality and its 5-95 percentiles (shaded region) as a function of time for $\sigma_z = 1$.

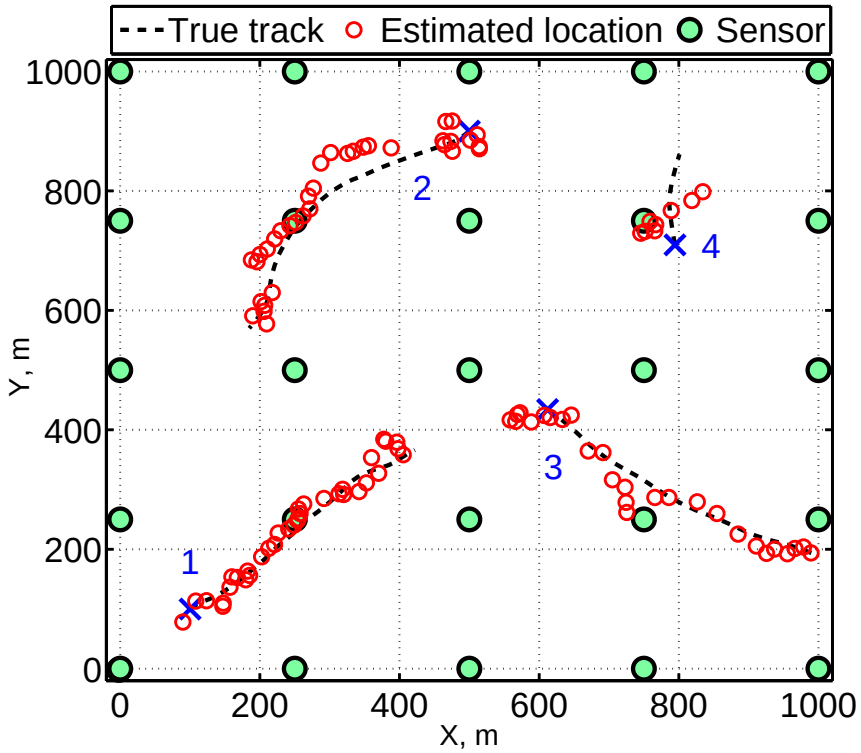


Fig. 5.10 Acoustic sensor network: True target tracks and estimated target locations obtained using the MBR-CPHD filter for $\sigma_{\mathbf{z}} = 1$.

Table 5.1 Average computational time required to process one observation vector (in seconds) for $\sigma_{\mathbf{z}} = 1$.

	Average computational time (seconds)	
Algorithm	RF tomography	Acoustic network
CPHD	3.76	0.73
MBR	2.54	0.63
MBR-CPHD	2.41	0.58

5.5.5 Computational requirements

Table 5.1 compares the average computational time required for the different algorithms to process one observation vector. All the simulations were performed using algorithms implemented in Matlab running on computers with two Xeon 4-core 2.5GHz processors and 14GB RAM. The time required to process one observation vector of the acoustic sensor network is much smaller than compared to that of the radio frequency sensor network because the measurement dimension is much smaller and there are fewer targets. The hybrid

MBR-CPHD filter is the fastest of the filters. The CPHD filter has higher computational requirements because of the costly clustering step required at each time step and the MBR filter has higher computational requirements because of the multiple additional particle filters employed to account for new target arrivals. The hybrid MBR-CPHD filter saves computation by initiating new particle filters only if the IIDC component indicates arrival of new targets and the costly clustering step is required only when multiple new targets arrive within the monitoring region in the same time step.

5.6 Conclusions

We studied the multi-Bernoulli filter and the hybrid multi-Bernoulli CPHD filter for the superpositional sensor model in this chapter. The update equation derivation methodology is similar for both the filters and is based on propagating the PHD of individual random finite set components. The cardinality distribution is additionally propagated for the hybrid multi-Bernoulli CPHD filter. We proposed auxiliary particle filter implementations of the filters and conducted a numerical study using a simulated RF tomography setup and an acoustic sensor network setup to perform multitarget tracking. The hybrid multi-Bernoulli CPHD filter performed better than the multi-Bernoulli filter and better than or as well as the CPHD filter when using the OSPA error metric. The hybrid filter was the least computationally demanding.

Chapter 6

Multisensor CPHD filter

In the previous chapters of this thesis we focused on superpositional sensors. In this chapter we turn our attention to multitarget tracking using standard sensors. A detailed description of the standard sensor model is given in Section 2.4.1. Many of the random finite set based filters such as the PHD filter [86], CPHD filter [87], and multi-Bernoulli filter [34, 39] were first proposed for the case of standard sensors. But the majority of the research has focused on developing filters where only one sensor is gathering measurements.

Several approximate as well as exact filters have been proposed that process measurements from multiple sensors. But most of them suffer from drawbacks such as sensor order dependence, numerical instability or high computational requirements. For example, the *iterated-corrector PHD filter* [127] processes the measurements from different sensors in a sequential manner but has a strong sensor order dependence. The *approximate product multisensor PHD and CPHD filters*, though independent of the sensor order, are numerically unstable and the problem worsens as the number of sensors increases. The exact *general multisensor PHD filter* was derived by Mahler [19] for the case of two sensors and generalized to include an arbitrary number of sensors by Delande et al. [122]. The exact filter update equations of the general multisensor PHD filter are not computationally tractable except for a few simple cases because of their combinatorial nature.

In this chapter we derive the update equations for the general multisensor CPHD filter. The derivation method is similar to that of the general multisensor PHD filter [19, 122] with the additional propagation of the cardinality distribution along with the probability hypothesis density function. Similar to the general multisensor PHD filter it has combina-

torial complexity and an exact implementation is computationally infeasible. To overcome this limitation we propose a two-step greedy approach based on a Gaussian mixture model implementation. Each step can be realized using a trellis structure constructed using the measurements from different sensors or measurement subsets for different Gaussian components. The algorithm is applicable to both the general multisensor CPHD and the general multisensor PHD filters.

Organization of this chapter is as follows. In Section 6.1 we provide a brief review of the existing multisensor filters. In Section 6.2 we derive the update equations of the general multisensor CPHD filter. We discuss a computationally tractable implementation of the general multisensor PHD and CPHD filters in Section 6.3. A performance comparison of the proposed filter with existing multisensor filters is done using numerical simulations in Section 6.4. We provide conclusions in Section 6.5.

6.1 Review of multisensor filters

We will briefly review the iterated-corrector PHD/CPHD filters and the approximate product PHD/CPHD filters in this section. The general multisensor PHD filter is derived as a special case of the general multisensor CPHD filter in Section 6.2.2. Different filters differ in the way observations are processed hence the update step is different. The prediction steps are the same as those discussed in Sections 4.2.1 and 4.3.1. Hence we do not discuss the prediction step of the various filters in this chapter.

The standard sensor observation process is discussed in Section 2.4.1. Assume that there are s sensors, and conditional on the multitarget state, their observations are independent. Measurements $\mathbf{z}^j$ gathered by sensor j lie in the space $\mathcal{Z}^j$, i.e., $\mathbf{z}^j \in \mathcal{Z}^j$. Let $Z_k^j = \{\mathbf{z}_{1,k}^j, \mathbf{z}_{2,k}^j, \ldots, \mathbf{z}_{m_{j,k}}^j\}$, $Z_k^j \subseteq \mathcal{Z}^j$ be the set of measurements collected by the j-th sensor at time step k. The measurement set can be empty. We assume that each target generates at most one measurement per sensor at each time instant k. Each measurement is either associated with a target or is generated by the clutter process. Define $Z_k^{1:s} = \{Z_k^1, Z_k^2, \ldots, Z_k^s\}$ to be the collection of measurement sets gathered by all sensors at time k. The function $h_{j,k}(\mathbf{z}|\mathbf{x})$ denotes the probability density (likelihood) that sensor j makes a measurement $\mathbf{z}$ given that it detects a target with state $\mathbf{x}$. Let $\mathcal{X}$ be the single target state space.

The various multisensor PHD filters make the following modelling assumptions to derive the update equations:

Assumption 1:

a) The predicted multitarget state is at time $k+1$ is modelled using a Poisson random finite set (Section 2.3.1).

b) Given a target with state $\mathbf{x}$, the probability that sensor j detects the target is given by $p^j_{d,k}(\mathbf{x})$. Denote the probability of a missed detection as $q^j_{d,k}(\mathbf{x}) = 1 - p^j_{d,k}(\mathbf{x})$.

c) The target-generated sensor measurements are independent conditional on the multitarget state X_{k+1}.

d) The sensor clutter processes are assumed to be Poisson. Let $c_{k+1,j}(\mathbf{z})$ be the clutter spatial distribution and $\lambda_{j,k+1}$ be the clutter rate of the j^{th} sensor.

Similarly, the various multisensor CPHD filters make the following modelling assumptions to derive the update equations:

Assumption 2:

a) The predicted multitarget state at time $k+1$ is modelled using an IIDC random finite set (Section 2.3.2). Let $\pi_{k+1|k}(n)$ be the predicted cardinality distribution at time $k+1$ and $M_{k+1|k}(y)$ be the probability generating function of $\pi_{k+1|k}(n)$.

b) Given a target with state $\mathbf{x}$, the probability that sensor j detects the target is given by $p^j_{d,k}(\mathbf{x})$.

c) The target-generated sensor measurements are independent conditional on the multitarget state X_{k+1}.

d) The sensor clutter processes are assumed to be IIDC. Let $c_{k+1,j}(\mathbf{z})$ be the clutter spatial distribution and $C_{k+1,j}(y)$ be the PGF of the clutter cardinality distribution of the j^{th} sensor.

Denote the v^{th}-order derivatives of the PGFs of the clutter cardinality distribution and the predicted cardinality distribution as

$$C^{(v)}_j(y) = \frac{d^v C_j}{dy^v}(y)\,, \quad M^{(v)}(y) = \frac{d^v M_{k+1|k}}{dt^v}(y). \tag{6.1}$$

Let $\sigma_{m,i}(y_1, \ldots, y_m)$ denote the elementary symmetric function of degree i in m variables defined as

$$\sigma_{m,i}(y_1, \ldots, y_m) = \sum_{1 \leq j_1 < j_2 \cdots < j_i \leq m} y_{j_1} y_{j_2} \ldots y_{j_i}. \tag{6.2}$$

Let the predicted PHD at time step $k+1$ be $D_{k+1|k}(\mathbf{x})$ and $r_{k+1|k}(\mathbf{x})$ denote the normalized predicted PHD function. Also let $\mu_{k+1|k}$ be the average predicted cardinality.

$$r_{k+1|k}(\mathbf{x}) = \frac{D_{k+1|k}(\mathbf{x})}{\mu_{k+1|k}(\mathbf{x})} \tag{6.3}$$

$$\mu_{k+1|k} = \int_{\mathcal{X}} D_{k+1|k}(\mathbf{x}) d\mathbf{x} = \sum_{n \geq 0} n \, \pi_{k+1|k}(n) \tag{6.4}$$

To keep the expressions compact we drop the time index and denote

$$c_{k+1,j}(\mathbf{z}) \equiv c_j(\mathbf{z}), \quad C_{k+1,j}(y) \equiv C_j(y), \quad p^j_{d,k+1}(\mathbf{x}) \equiv p^j_d(\mathbf{x}) \tag{6.5}$$

$$M_{k+1|k}(y) \equiv M(y), \quad q^j_{d,k+1}(\mathbf{x}) \equiv q^j_d(\mathbf{x}), \quad r_{k+1|k}(\mathbf{x}) \equiv r(\mathbf{x}) \tag{6.6}$$

$$h_{j,k+1}(\mathbf{z}|\mathbf{x}) \equiv h_j(\mathbf{z}|\mathbf{x}), \qquad m^j_{k+1} \equiv m^j, \qquad \lambda_{j,k+1} \equiv \lambda_j, \tag{6.7}$$

when the time is clear from the context. Note that abbreviated notation is used only for convenience and the above quantities are in general functions of time. We use the notation $[\![1, s]\!]$ to denote the set of integers from 1 to s.

6.1.1 Iterated-corrector PHD and CPHD filters

The iterated-corrector PHD and CPHD filters process multisensor information in a sequential manner. A single sensor PHD filter processes measurements from the first sensor. Using the output PHD function produced by this step as the predicted PHD function, another single sensor PHD filter processes measurements from the second sensor and so on. In the iterated-corrector CPHD filter along the the PHD function, each of the single sensor CPHD filters also propagates the cardinality distribution to the subsequent single sensor CPHD filter.

Iterated-corrector PHD filter

Let $D^{[j]}_{k+1|k+1}(\mathbf{x})$ be the output PHD after processing the observation set Z^j_{k+1} from the j^{th} sensor. Also denote $D^{[0]}_{k+1|k+1}(\mathbf{x}) = D_{k+1|k}(\mathbf{x})$. Then the update equations for the iterated-corrector PHD filter for $j \in [\![1, s]\!]$ are

$$D^{[j]}_{k+1|k+1}(\mathbf{x}) = L^{[j]}_{Z^j_{k+1}}(\mathbf{x})\, D^{[j-1]}_{k+1|k+1}(\mathbf{x}) \tag{6.8}$$

$$L^{[j]}_{Z^j_{k+1}}(\mathbf{x}) \stackrel{\text{def}}{=} q^j_d(\mathbf{x}) + \sum_{\mathbf{z}\in Z^j_{k+1}} \frac{p^j_d(\mathbf{x})\, h_j(\mathbf{z}|\mathbf{x})}{\lambda_j c_j(\mathbf{z}) + \langle D^{[j-1]}_{k+1|k+1}, p^j_d\, h_j(\mathbf{z})\rangle} \tag{6.9}$$

Equation (6.8) is the single sensor PHD update equation [86]. The PHD obtained after processing all the s sensors is the final posterior PHD of the iterated-corrector PHD filter $D_{k+1|k+1}(\mathbf{x}) = D^{[s]}_{k+1|k+1}(\mathbf{x})$.

Iterated-corrector CPHD filter

Let $D^{[j]}_{k+1|k+1}(\mathbf{x})$ and $\pi^{[j]}_{k+1|k+1}(n)$ be the PHD and cardinality distribution after processing the j^{th} sensor. Let the normalized PHDs be denoted by $r^{[j]}_{k+1|k+1}(\mathbf{x})$

$$r^{[j]}_{k+1|k+1}(\mathbf{x}) = \frac{D^{[j]}_{k+1|k+1}(\mathbf{x})}{\int_{\mathcal{X}} D^{[j]}_{k+1|k+1}(\mathbf{x})d\mathbf{x}} \tag{6.10}$$

Denote $D^{[0]}_{k+1|k+1}(\mathbf{x}) = D_{k+1|k}(\mathbf{x})$, $r^{[0]}_{k+1|k+1}(\mathbf{x}) = r_{k+1|k}(\mathbf{x})$ and $\pi^{[0]}_{k+1|k+1}(n) = \pi_{k+1|k}(n)$. We also define the following quantities

$$\gamma_{[j]} \stackrel{\text{def}}{=} \langle r^{[j]}_{k+1|k+1}, q^j_d\rangle \tag{6.11}$$

$$\sigma^{[j]}_i(Z^j_{k+1}) \stackrel{\text{def}}{=} \sigma_{m^j,i}\left(\frac{\langle r^{[j]}_{k+1|k+1}, p^j_d\, h_j(\mathbf{z}^j_1)\rangle}{c_j(\mathbf{z}^j_1)}, \ldots, \frac{\langle r^{[j]}_{k+1|k+1}, p^j_d\, h_j(\mathbf{z}^j_{m^j})\rangle}{c_j(\mathbf{z}^j_{m^j})}\right) \tag{6.12}$$

$$\alpha_0^{[j]} \stackrel{\text{def}}{=} \frac{\sum_{i=0}^{m^j} C_j^{(m^j-i)}(0)\, G^{(i+1)}(\gamma_{[j]})\, \sigma_i^{[j]}(Z_{k+1}^j)}{\sum_{i=0}^{m^j} C_j^{(m^j-i)}(0)\, G^{(i)}(\gamma_{[j]})\, \sigma_i^{[j]}(Z_{k+1}^j)} \tag{6.13}$$

$$\alpha_{\mathbf{z}}^{[j]} \stackrel{\text{def}}{=} \frac{\sum_{i=0}^{m^j-1} C_j^{(m^j-i-1)}(0)\, G^{(i+1)}(\gamma_{[j]})\, \sigma_i^{[j]}(Z_{k+1}^j - \mathbf{z})}{\sum_{i=0}^{m^j} C_j^{(m^j-i)}(0)\, G^{(i)}(\gamma_{[j]})\, \sigma_i^{[j]}(Z_{k+1}^j)} \tag{6.14}$$

The update equations for the iterated-corrector CPHD filter are then given by

$$D_{k+1|k+1}^{[j]}(\mathbf{x}) = L_{Z_{k+1}^j}^{[j]}(\mathbf{x})\, D_{k+1|k+1}^{[j-1]}(\mathbf{x}) \tag{6.15}$$

$$\pi_{k+1|k+1}^{[j]}(n) = \frac{\ell_{Z_{k+1}^j}^{[j]}(n)\pi_{k+1|k+1}^{[j-1]}(n)}{\sum_{i\geq 0} \ell_{Z_{k+1}^j}^{[j]}(i)\pi_{k+1|k+1}^{[j-1]}(i)} \tag{6.16}$$

where

$$L_{Z_{k+1}^j}^{[j]}(\mathbf{x}) \stackrel{\text{def}}{=} \alpha_0^{[j]}(q_d^j(\mathbf{x})) + \sum_{\mathbf{z}\in Z_{k+1}^j} \frac{p_d^j(\mathbf{x})\, h_j(\mathbf{z}|\mathbf{x})\, \alpha_{\mathbf{z}}^{[j]}}{c_j(\mathbf{z})} \tag{6.17}$$

$$\ell_{Z_{k+1}^j}^{[j]}(n) \stackrel{\text{def}}{=} \sum_{i=0}^{\min(n,m^j)} C_j^{(m^j-i)}(0)\, \frac{n!}{(n-i)!}\, \gamma_{[j]}^{n-i}\, \sigma_i^{[j]}(Z_{k+1}^j) \tag{6.18}$$

6.1.2 Approximate product PHD and CPHD filters

The approximate product filters developed by Mahler process the multisensor information in a single update step. Their derivation is based on the assumption that the multitarget distribution $f_{k+1|k+1}(X|Z_{k+1,j})$, i.e., the distribution obtained by updating only the j^{th} sensor, corresponds to an IIDC process for each sensor $j \in [\![1, s]\!]$. The filters are called product filters because the combined multisensor pseudo-likelihood function is a product of the single sensor likelihoods with an additional scaling factor. We define the following quantities for use in update equations of both the approximate product PHD and CPHD

filters

$$\gamma_j \stackrel{\text{def}}{=} \langle r_{k+1|k}, q_d^j \rangle \tag{6.19}$$

$$\sigma_i^j(Z_{k+1}^j) \stackrel{\text{def}}{=} \sigma_{m_j,i}\left(\frac{\langle r_{k+1|k}, p_d^j\, h_j(\mathbf{z}_1^j)\rangle}{c_j(\mathbf{z}_1^j)}, \ldots, \frac{\langle r_{k+1|k}, p_d^j\, h_j(\mathbf{z}_{m^j}^j)\rangle}{c_j(\mathbf{z}_{m^j}^j)} \right) \tag{6.20}$$

Approximate product PHD filter

We associate the following quantities with the j^{th} sensor

$$L^j_{Z^j_{k+1}}(\mathbf{x}) \stackrel{\text{def}}{=} q_d^j(\mathbf{x}) + \sum_{\mathbf{z}\in Z_{k+1,j}} \frac{p_d^j(\mathbf{x})\, h_j(\mathbf{z}|\mathbf{x})}{\lambda_j\, c_j(\mathbf{z}) + \langle D_{k+1|k}, p_d^j\, h_j(\mathbf{z})\rangle} \tag{6.21}$$

$$N_{k+1}^j \stackrel{\text{def}}{=} \langle r_{k+1|k}, L^j_{Z^j_{k+1}} \rangle \tag{6.22}$$

$$\ell^j_{Z^j_{k+1}}(n) \stackrel{\text{def}}{=} \sum_{i=0}^{\min(n,m_j)} \lambda_j^{m_j-i} \frac{n!}{(n-i)!} \gamma_j^{n-i} \sigma_i^j(Z_{k+1}^j) \tag{6.23}$$

Note that $L^j_{Z^j_{k+1}}(\mathbf{x})$ is the pseudo-likelihood function associated with the single sensor PHD filter processing data from the j^{th} sensor and N_{k+1}^j is the corresponding mean of the posterior cardinality. Also define

$$\omega \stackrel{\text{def}}{=} \frac{\int_{\mathcal{X}} r_{k+1|k}(\mathbf{x}) \prod_{j=1}^{s} L^j_{Z^j_{k+1}}(\mathbf{x}) d\mathbf{x}}{\prod_{j=1}^{s} N_{k+1}^j}, \quad \phi \stackrel{\text{def}}{=} \frac{\sum_{i\geq 0} \frac{(N_{k+1|k}\omega)^i}{i!} \prod_{j=1}^{s} \ell^j_{Z^j_{k+1}}(i+1)}{\sum_{i\geq 0} \frac{(N_{k+1|k}\omega)^i}{i!} \prod_{j=1}^{s} \ell^j_{Z^j_{k+1}}(i)} \tag{6.24}$$

Then the PHD update equation for the approximate product multisensor PHD filter and the multisensor pseudo-likelihood function are given as

$$D_{k+1|k+1}(\mathbf{x}) = L_{Z^{1:s}_{k+1}}(\mathbf{x})\, D_{k+1|k}(\mathbf{x}), \quad L_{Z^{1:s}_{k+1}}(\mathbf{x}) \stackrel{\text{def}}{=} \phi \frac{\prod_{j=1}^{s} L^j_{Z^j_{k+1}}(\mathbf{x})}{\prod_{j=1}^{s} N_{k+1}^j} \tag{6.25}$$

The multisensor pseudo-likelihood function $L_{Z^{1:s}_{k+1}}(\mathbf{x})$ is the scaled product of the individual single sensor pseudo-likelihood functions $L^j_{Z^j_{k+1}}(\mathbf{x})$.

Approximate product CPHD filter

We define the following quantities associated with the j^{th} sensor

$$\alpha_0^j \stackrel{\text{def}}{=} \frac{\sum_{i=0}^{m_j} C_j^{(m_j-i)}(0)\, G^{(i+1)}(\gamma_j)\, \sigma_i^j(Z_{k+1}^j)}{\sum_{i=0}^{m_j} C_j^{(m_j-i)}(0)\, G^{(i)}(\gamma_j)\, \sigma_i^j(Z_{k+1}^j)} \tag{6.26}$$

$$\alpha_{\mathbf{z}}^j \stackrel{\text{def}}{=} \frac{\sum_{i=0}^{m_j-1} C_j^{(m_j-i-1)}(0) G^{(i+1)}(\gamma_j)\, \sigma_i^j(Z_{k+1}^j - \{\mathbf{z}\})}{\sum_{i=0}^{m_j} C_j^{(m_j-i)}(0)\, G^{(i)}(\gamma_j)\, \sigma_i^j(Z_{k+1}^j)} \tag{6.27}$$

$$L_{Z_{k+1}^j}^j(\mathbf{x}) \stackrel{\text{def}}{=} \alpha_0^j\, q_d^j(\mathbf{x}) + \sum_{\mathbf{z}\in Z_{k+1}^j} \frac{p_d^j(\mathbf{x})\, h_j(\mathbf{z}|\mathbf{x})\alpha_{\mathbf{z}}^j}{c_j(\mathbf{z})} \tag{6.28}$$

$$N_{k+1}^j \stackrel{\text{def}}{=} \langle r_{k+1|k}, L_{Z_{k+1}^j}^j \rangle \tag{6.29}$$

$$\ell_{Z_{k+1}^j}^j(n) \stackrel{\text{def}}{=} \sum_{i=0}^{\min(n,m_j)} \frac{n!}{(n-i)!} C_j^{(m_j-i)}(0)\, \gamma_j^{n-i} \sigma_i^j(Z_{k+1}^j). \tag{6.30}$$

Note that the above quantities $L_{Z_{k+1}^j}^j(\mathbf{x})$ and $\ell_{Z_{k+1}^j}^j(n)$ are the pseudo-likelihoods associated with the single sensor CPHD filter processing measurements from the j^{th} sensor. Also define

$$\omega \stackrel{\text{def}}{=} \frac{\int_{\mathcal{X}} r_{k+1|k}(\mathbf{x}) \prod_{j=1}^{s} L_{Z_{k+1}^j}^j(\mathbf{x}) d\mathbf{x}}{\prod_{j=1}^{s} N_{k+1}^j}, \quad \phi \stackrel{\text{def}}{=} \frac{\sum_{i\geq 0} \omega^i\, (i+1)\, p_{k+1|k}(i+1) \prod_{j=1}^{s} \ell_{Z_{k+1}^j}^j(i+1)}{\sum_{i\geq 0} \omega^i\, p_{k+1|k}(i) \prod_{j=1}^{s} \ell_{Z_{k+1}^j}^j(i)}. \tag{6.31}$$

The PHD update equation for the approximate product multisensor CPHD filter and the multisensor pseudo-likelihood function $L_{Z_{k+1}^{1:s}}(\mathbf{x})$ are given as

$$D_{k+1|k+1}(\mathbf{x}) = L_{Z_{k+1}^{1:s}}(\mathbf{x})\, r_{k+1|k}(\mathbf{x}) \tag{6.32}$$

$$L_{Z_{k+1}^{1:s}}(\mathbf{x}) \stackrel{\text{def}}{=} \phi \frac{\prod_{j=1}^{s} L^{j}_{Z^{j}_{k+1}}(\mathbf{x})}{\prod_{j=1}^{s} N^{j}_{k+1}}. \tag{6.33}$$

The cardinality update for the approximate product multisensor CPHD filter is

$$\pi_{k+1|k+1}(n) = \frac{\omega^n\, \pi_{k+1|k}(n) \prod_{j=1}^{s} \ell^{j}_{Z^{j}_{k+1}}(n)}{\sum_{i\geq 0} \omega^i\, \pi_{k+1|k}(i) \prod_{j=1}^{s} \ell^{j}_{Z^{j}_{k+1}}(i)} \tag{6.34}$$

6.2 General multisensor CPHD filter

In this section we outline the main result of the general multisensor CPHD filter update step. The derivation method is similar to the approach used to derive the general multisensor PHD filter equations by Mahler [19] and Delande et al. [122]. The prediction step is the same as that for the single sensor case and is discussed in Section 4.3.1.

6.2.1 CPHD update step

Let $W \subseteq Z_{k+1}^{1:s}$ such that for all $j \in [\![1, s]\!]$, $|W|_j \leq 1$ where $|W|_j = |\{\mathbf{z} \in W : \mathbf{z} \in Z_{k+1}^{j}\}|$. Thus the subset W can have at most one measurement from each sensor. Let $\mathcal{W}$ be the set of all such W. For any measurement subset W we can uniquely associate with it a set of pairs of indices T_W defined as $T_W = \{(j, l) : \mathbf{z}_l^j \in W\}$.

For disjoint subsets $W_1, W_2, \ldots W_n$, let $V = Z_{k+1}^{1:s} \setminus (\cup_{i=1}^{n} W_i)$, so that $W_1, W_2, \ldots W_n$ and V partition $Z_{k+1}^{1:s}$. We can think of the set W_i as a collection of measurements made by different sensors, all of which are generated by the same target and the set V as the collection of clutter measurements made by all the sensors. Let P be a partition of $Z_{k+1}^{1:s}$, constructed

using elements from the set $\mathcal{W}$ and a set V, given by

$$P = \{W_1, W_2, \ldots W_{|P|-1}, V\}, \tag{6.35}$$

such that

$$\bigcup_{i=1}^{|P|-1} W_i \cup V = Z_{k+1}^{1:s}, \tag{6.36}$$

$$W_i \cap W_j = \varnothing, \text{for any } W_i, W_j \in P, i \neq j \tag{6.37}$$

$$W_i \cap V = \varnothing, \text{for any } W_i \in P \tag{6.38}$$

where $|P|$ denotes the number of elements in the partition P.

The partition P groups the measurements in $Z_{k+1}^{1:s}$ into disjoint subsets where each subset is either generated by a target (the W subsets) or generated by the clutter process (the V subset). Let $|P|_j$ be the number of measurements made by sensor j which are generated by the targets and is given by

$$|P|_j = \sum_{i=1}^{|P|-1} |W_i|_j. \tag{6.39}$$

The number of measurements made by sensor j which are classified as clutter in the partition P is $(m_j - |P|_j)$. Let $\mathcal{P}$ be the collection of all possible partitions P of $Z_{k+1}^{1:s}$ constructed as above. A recursive expression for constructing the collection $\mathcal{P}$ is given in Appendix C.1.

Denote the v^{th}-order derivatives of the PGFs of the clutter cardinality distribution and the predicted cardinality distribution as

$$C_j^{(v)}(t) = \frac{d^v C_j}{dt^v}(t), \quad M^{(v)}(t) = \frac{d^v M}{dt^v}(t). \tag{6.40}$$

We use γ to denote the probability, under the predictive PHD, that a target is detected by no sensor, and we thus have:

$$\gamma \stackrel{\text{def}}{=} \int_{\mathcal{X}} r(\mathbf{x}) \prod_{j=1}^{s} q_d^j(\mathbf{x}) d\mathbf{x}. \tag{6.41}$$

For concise specification of the update equations, it is useful to combine the terms associated with the PGF of the clutter cardinality distribution for a partition P. Let us

define the quantity

$$\kappa_P = \prod_{j=1}^{s} C_j^{(m_j - |P|_j)}(0). \tag{6.42}$$

For a set $W \in \mathcal{W}$ and the associated index set T_W define the quantities

$$d_W \stackrel{\text{def}}{=} \frac{\int_{\mathcal{X}} r(\mathbf{x}) \left(\prod_{(i,l)\in T_W} p_d^i(\mathbf{x})\, h_i(\mathbf{z}_l^i|\mathbf{x}) \right) \prod_{j:(j,*)\notin T_W} q_d^j(\mathbf{x}) d\mathbf{x}}{\prod_{(i,l)\in T_W} c_i(\mathbf{z}_l^i)}, \tag{6.43}$$

$$\rho_W(\mathbf{x}) \stackrel{\text{def}}{=} \frac{\left(\prod_{(i,l)\in T_W} p_d^i(\mathbf{x})\, h_i(\mathbf{z}_l^i|\mathbf{x}) \right) \prod_{j:(j,*)\notin T_W} q_d^j(\mathbf{x})}{\int_{\mathcal{X}} r(\mathbf{x}) \left(\prod_{(i,l)\in T_W} p_d^i(\mathbf{x})\, h_i(\mathbf{z}_l^i|\mathbf{x}) \right) \prod_{j:(j,*)\notin T_W} q_d^j(\mathbf{x}) d\mathbf{x}}, \tag{6.44}$$

where $(j,*)$ indicates any pair of indices of the form (j,l).

The updated probability hypothesis density function $D_{k+1|k+1}(\mathbf{x})$ can be expressed as the product of the normalized predicted probability hypothesis density $r_{k+1|k}(\mathbf{x})$ at time $k+1$ and a pseudolikelihood function. The pseudolikelihood function can be expressed as a linear combination of functions (one function for each partition P) with associated weights α_P. The all-clutter partition $P = \{V\}$ where $V = Z_{k+1}^{1:s}$ is not included in the collection $\mathcal{P}$ and has an associated weight α_0. Define

$$\alpha_0 \stackrel{\text{def}}{=} \frac{\sum_{P\in\mathcal{P}} \left(\kappa_P M^{(|P|)}(\gamma) \prod_{W\in P} d_W \right)}{\sum_{P\in\mathcal{P}} \left(\kappa_P M^{(|P|-1)}(\gamma) \prod_{W\in P} d_W \right)}, \tag{6.45}$$

$$\alpha_P \stackrel{\text{def}}{=} \frac{\kappa_P M^{(|P|-1)}(\gamma) \prod_{W\in P} d_W}{\sum_{Q\in\mathcal{P}} \left(\kappa_Q M^{(|Q|-1)}(\gamma) \prod_{W\in Q} d_W \right)}. \tag{6.46}$$

Note that the expression $W \in P$ only includes $W \in \mathcal{W}$ and does not include the component $V \in \mathcal{V}$ of P.

Theorem 1. *Under the conditions of Assumption 2, the general multisensor CPHD filter update equation for the probability hypothesis density is*

$$\frac{D_{k+1|k+1}(\mathbf{x})}{r_{k+1|k}(\mathbf{x})} = \alpha_0 \prod_{j=1}^{s} q_d^j(\mathbf{x}) + \sum_{P\in\mathcal{P}} \alpha_P \left(\sum_{W\in P} \rho_W(\mathbf{x}) \right) \tag{6.47}$$

and the update equation for the posterior cardinality distribution is

$$\frac{\pi_{k+1|k+1}(n)}{\pi_{k+1|k}(n)} = \frac{\sum_{\substack{P\in\mathcal{P} \\ |P|\leq n+1}} \left(\kappa_P \frac{n!}{(n-|P|+1)!} \gamma^{n-|P|+1} \prod_{W\in P} d_W \right)}{\sum_{P\in\mathcal{P}} \left(\kappa_P M^{(|P|-1)}(\gamma) \prod_{W\in P} d_W \right)}, \tag{6.48}$$

where the quantities α_0, α_P, $\rho_W(\mathbf{x})$ and d_W are given in (6.45), (6.46), (6.44) *and* (6.43).

The proof of Theorem 1 is provided in Appendix C.5. It requires the concepts of functional derivatives and integral transform of the posterior multitarget density which are revised in Appendices C.2 and C.3 respectively. The proof depends on an intermediate result, Lemma 1, which is proved in Appendix C.4.

6.2.2 General multisensor PHD filter as a special case

In this section we show that the general multisensor PHD filter can be obtained as a special case of the general multisensor CPHD filter. The assumptions made by the PHD filter are given in Assumption 1 in Section 6.1. Since the multitarget state distribution is modelled as Poisson it suffices to propagate the PHD function over time. Let the rate of the Poisson clutter process be λ_j and let $c_j(\mathbf{z})$ be the clutter spatial distribution for the j^{th} sensor. Let $\mu_{k+1|k}$ be the mean predicted cardinality at time $k+1$. Using the Poisson assumptions for the predicted multitarget distribution and the sensor clutter processes we have

$$M^{(v)}(\gamma) = \mu_{k+1|k}^v \, e^{\mu_{k+1|k}(\gamma-1)} \tag{6.49}$$

$$C_j^{(v)}(0) = \lambda_j^v \, e^{-\lambda_j}. \tag{6.50}$$

Using these in (6.45) we have the simplification $\alpha_0 = \mu_{k+1|k}$. We can also simplify the term $\kappa_P M^{(|P|-1)}(\gamma)$ as

$$\kappa_P M^{(|P|-1)}(\gamma) = \left(\prod_{j=1}^{s} C_j^{(m_j-|P|_j)}(0)\right) M^{(|P|-1)}(\gamma) \tag{6.51}$$

$$= \left(\prod_{j=1}^{s} \lambda_j^{m_j-|P|_j} e^{-\lambda_j}\right) \mu_{k+1|k}^{|P|-1} e^{\mu_{k+1|k}(\gamma-1)} \tag{6.52}$$

$$= \left(e^{\mu_{k+1|k}(\gamma-1)-\sum_{j=1}^{s}\lambda_j}\right)\left(\prod_{j=1}^{s}\lambda_j^{m_j}\right)\frac{\mu_{k+1|k}^{|P|-1}}{\left(\prod_{j=1}^{s}\lambda_j^{|P|_j}\right)}. \tag{6.53}$$

Since the expression $\kappa_P M^{(|P|-1)}(\gamma)$ appears in both the numerator and the denominator of the term α_P in the PHD update expression, we can ignore the portion that is independent of P. Hence we have

$$\kappa_P M^{(|P|-1)}(\gamma) \propto \mu_{k+1|k}^{|P|-1} \prod_{j=1}^{s} \lambda_j^{-|P|_j}. \tag{6.54}$$

From (6.43) and (6.54), we can write

$$\kappa_P M^{(|P|-1)}(\gamma) \prod_{W\in P} d_W \propto \prod_{W\in P} \tilde{d}_W \tag{6.55}$$

where $\tilde{d}_W$ is defined as

$$\tilde{d}_W \stackrel{\text{def}}{=} \frac{\mu_{k+1|k}\int_{\mathcal{X}} r(\mathbf{x})\left(\prod_{(i,l)\in T_W} p_d^i(\mathbf{x})\, h_i(\mathbf{z}_l^i|\mathbf{x})\right)\prod_{j:(j,*)\notin T_W} q_d^j(\mathbf{x})d\mathbf{x}}{\prod_{(i,l)\in T_W}\lambda_i\, c_i(\mathbf{z}_l^i)}. \tag{6.56}$$

The PHD update equation then reduces to

$$\frac{D_{k+1|k+1}(\mathbf{x})}{r_{k+1|k}(\mathbf{x})} = \mu_{k+1|k}\prod_{j=1}^{s} q_d^j(\mathbf{x}) + \sum_{P\in\mathcal{P}} \frac{\left(\prod_{W\in P}\tilde{d}_W\right)\sum_{W\in P}\rho_W(\mathbf{x})}{\sum_{P\in\mathcal{P}}\left(\prod_{W\in P}\tilde{d}_W\right)}. \tag{6.57}$$

This equation is equivalent to the general multisensor PHD update equation discussed in the literature [19, 122].

6.3 Approximate implementation of the general multisensor CPHD and PHD filters

In the previous section we derived update equations for the general multisensor CPHD filter which propagate the PHD and cardinality distribution over time. Analytic propagation of these quantities is difficult in general without imposing further conditions. In Section 6.3.1 we develop a Gaussian mixture-based implementation of the filter update equations. Although the Gaussian mixture implementation is analytically tractable, it is computationally intractable. In Sections 6.3.2 and 6.3.3 we propose greedy algorithms to drastically reduce computations and develop computationally tractable approximate implementations for the general multisensor CPHD and PHD filters.

6.3.1 Gaussian mixture implementation

We make the following assumptions to obtain closed form updates for equations (6.47) and (6.48)

Assumption 3:

a) The probability of detection for each sensor is constant throughout the single target state space; i.e., $p_d^j(\mathbf{x}) = p_d^j$, for all $\mathbf{x}$.

b) The predicted PHD is a mixture of weighted Gaussian densities.

c) The single sensor observations are linear functions of a single target state corrupted by zero-mean Gaussian noise.

d) The predicted cardinality distribution has finite support; i.e., there exists a positive integer $n_0 < \infty$ such that $\pi_{k+1|k}(n) = 0$, for all $n > n_0$.

From the above assumptions we can express the normalized predicted PHD as a Gaussian mixture model

$$r(\mathbf{x}) = \sum_{i=1}^{J_{k+1|k}} w_{k+1|k}^{(i)} \mathcal{N}^{(i)}(\mathbf{x}) \tag{6.58}$$

$$\mathcal{N}^{(i)}(\mathbf{x}) \stackrel{\text{def}}{=} \mathcal{N}(\mathbf{x}; \mathbf{m}_{k+1|k}^{(i)}, \Sigma_{k+1|k}^{(i)}) \tag{6.59}$$

where $w_{k+1|k}^{(i)}$ are non-negative weights satisfying $\sum_{i=1}^{J_{k+1|k}} w_{k+1|k}^{(i)} = 1$; and $\mathcal{N}^{(i)}(\mathbf{x})$ is the Gaussian density function with mean $\mathbf{m}_{k+1|k}^{(i)}$ and covariance matrix $\Sigma_{k+1|k}^{(i)}$. If H_j is the observation matrix for sensor j then its likelihood function can be expressed as $h_j(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}; H_j\mathbf{x}, \Sigma_j)$. Then under the conditions of Assumption 3, the posterior PHD at time $k+1$ can be expressed as a weighted mixture of Gaussian densities and the posterior cardinality distribution has a finite support.

Since the probability of detection is constant we have $\gamma = \prod_{j=1}^{s} q_d^j$. For each partition P the quantities $M^{(|P|-1)}$ and $M^{(|P|)}$ can be easily calculated since the predicted cardinality distribution has finite support. The integration in the numerator of (6.43) is analytically solvable under Assumption 3 and using properties of Gaussian density functions [41]. Hence d_W can be analytically evaluated. From these quantities we can calculate α_0 and α_P from (6.45) and (6.46). For each measurement set W we can express the product $r(\mathbf{x})\rho_W(\mathbf{x})$ as a sum of weighted Gaussian densities using the properties of Gaussian density functions [41]. Thus from the update equation (6.47) the posterior PHD can be expressed as a mixture of Gaussian densities. Since the predicted cardinality distribution has finite support, from (6.48), the posterior cardinality distribution also has finite support. Similarly, under appropriate linear Gaussian assumptions, the posterior PHD in (6.57) can be expressed as a mixture of Gaussian densities.

Note that for deriving analytically tractable filter equations we make the simplifying assumption that the probability of detection $p_d(\mathbf{x})$ is a constant and does not depend on the target state $\mathbf{x}$. This requirement is not strictly necessary but is adopted in this thesis for simplicity. For a more realistic approximation, following the discussion in [41] we can relax this condition and model the probability of detection as a mixture of Gaussian densities and we would still obtain analytically tractable filter equations.

The conditions of Assumption 3 allow us to analytically propagate the PHD and cardinality distribution but the propagation is still numerically infeasible. The combinatorial

nature of the update step can be seen from (6.47), (6.48) and (6.57). Specifically, the exact implementation of the general multisensor CPHD and PHD filters would require evaluation of all the permissible partitions (i.e., all $P \in \mathcal{P}$) that could be constructed from all possible measurement subsets. The number of such partitions is prohibitively large and a direct implementation is infeasible. We now discuss an approximation of the update step to overcome this limitation.

The key idea of the approximate implementation is to identify elements of the collection $\mathcal{P}$ which make a significant contribution to the update expressions. We propose the following two-step greedy approximation to achieve this within the Gaussian mixture framework. The first approximation step is to select a few measurement subsets W for each Gaussian component that are best explained by that component. The second approximation step is to greedily construct partitions of these subsets which are significant for the update step. The following subsections explain these two steps in detail.

6.3.2 Selecting the best measurement subsets

A measurement subset is any subset of the measurement set $Z_{k+1}^{1:s}$ such that it contains at most one measurement per sensor. The total number of measurement subsets that can be constructed when the j^{th} sensor records m_j measurements is $\prod_{j=1}^{s}(m_j+1)$. When there are many targets present and/or the clutter rate is high this number can be very large. Since the size of the collection $\mathcal{P}$ depends on the number of measurement subsets, to develop a tractable implementation of the update step it is necessary to limit the number of measurement subsets. We achieve this by associating a score with each measurement subset that correlates with its significance in the final update step. Instead of enumerating all possible measurement subsets, they are greedily and sequentially constructed and only a few are retained based on the scores associated with them.

Consider the measurement subset W and the associated set T_W as defined earlier. For the i^{th} Gaussian component and the measurement subset W we can associate a score

function $\beta^{(i)}(W)$ defined as

$$\beta^{(i)}(W) \stackrel{\text{def}}{=} \frac{\int_{\mathcal{X}} \mathcal{N}^{(i)}(\mathbf{x}) \left(\prod_{(j,l)\in T_W} p_d^j \, h_j(\mathbf{z}_l^j|\mathbf{x}) \right) \left(\prod_{j:(j,*)\notin T_W} q_d^j \right) d\mathbf{x}}{\prod_{(j,l)\in T_W} c_j(\mathbf{z}_l^j)}. \tag{6.60}$$

The above score function is obtained by splitting the d_W term in (6.43) for each Gaussian component. Intuitively, this score can be interpreted as the ratio of the likelihood that the measurement subset W was generated by the single target represented by the i^{th} Gaussian component to the likelihood that the measurement subset W was generated by the clutter process. The score $\beta^{(i)}(W)$ can be analytically calculated since the integral is solvable under Assumption 3 and using properties of Gaussian densities [36]. The score is high when the elements of the set W truly are the measurements caused by the target associated with the i^{th} Gaussian component. We use $\beta^{(i)}(W)$ to rank measurement subsets for each Gaussian component and retain only a fraction of them with the highest scores.

For each Gaussian component, we select the measurement subsets by randomly ordering the sensors and incrementally incorporating information from each sensor in turn. We retain a maximum of $W_{\max}$ subsets at each step. Figure 6.1 provides a graphical representation of the algorithm in the form of a trellis diagram. Each column of the trellis corresponds to observations from one of the sensors. The sensor number is indicated at top of each column. The nodes of the trellis correspond to the sensor observations $(\mathbf{z}_1^1, \mathbf{z}_2^1, \dots, \mathbf{z}_1^2, \mathbf{z}_2^2, \dots)$ or the no detection case $(\mathbf{z}_\varnothing^1, \mathbf{z}_\varnothing^2, \dots)$.

The process of sequential construction of measurement subsets can be demonstrated using an example as follows. The solid lines in Figure 6.1 represent partial measurement subsets retained after processing observations from sensors $1-3$. Now consider the measurement subset indicated by the thick solid line. It corresponds to the measurement subset $\{\mathbf{z}_1^2, \mathbf{z}_1^3\}$. When the sensor 4 measurements are processed, this measurement subset is extended for each node of sensor 4 as represented by the dashed lines. The scores $\beta^{(i)}(W)$ are calculated for these new measurement subsets using the expression in (6.60) but limited to only the first 4 sensors. This is done for each existing measurement subset in the sensor-measurement space and $W_{\max}$ measurement subsets with highest scores are retained and considered at the next sensor. Although the process of constructing measurement subsets is dependent on the order in which sensors are processed, we observe from simulations that

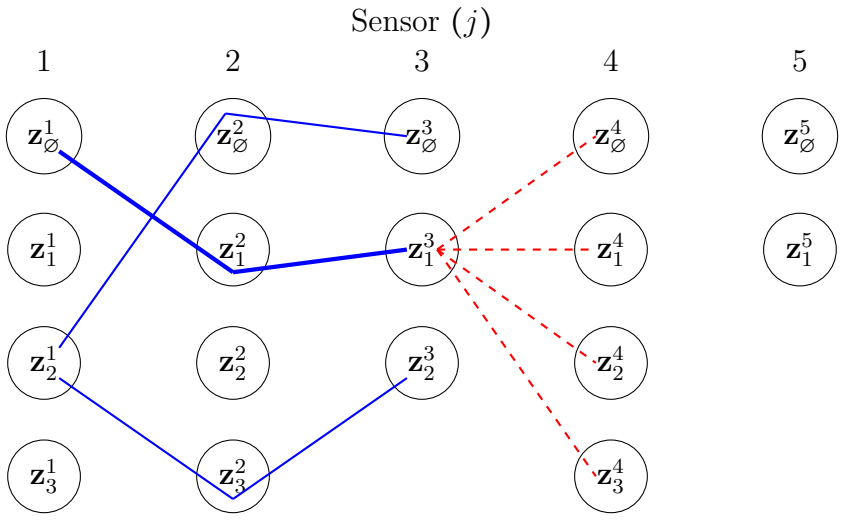


Fig. 6.1 Trellis diagram for constructing measurement subsets for each Gaussian component. Solid blue lines represent measurement subsets retained after processing sensors $1-3$. Dashed red lines correspond to extensions of retained measurement subsets when processing measurements from sensor 4.

it has no significant effect on filter performance. Once the subsets have been selected, the ordering has no further effect in the update process.

6.3.3 Constructing partitions

The algorithm to construct partitions from subsets is similar to the above algorithm used to identify the best measurement subsets. Since the V component of a partition is unique given the W components, it is sufficient to identify the W components to uniquely specify a partition P. A graphical representation of the algorithm is provided in Figure 6.2. Each column of this trellis corresponds to the set of measurement subsets $\{W^i_1, W^i_2, \dots\}$ identified by the i^{th} Gaussian component. The component number (i) is indicated at the top of each column. The node $W^i_\varnothing$ represents the empty measurement subset $W^i_\varnothing = \varnothing$ which is always included for each component and it corresponds to the event that the Gaussian component was not detected by any of the sensors. With each valid partition P we associate the score $d_P = \prod_{W \epsilon P} d_W$ with $d_\varnothing = 1$.

We greedily identify partitions of subsets by incrementally incorporating measurement subsets from the different components. For example the solid lines in Figure 6.2 correspond to the partitions that have been retained after processing components number $1--3$. The

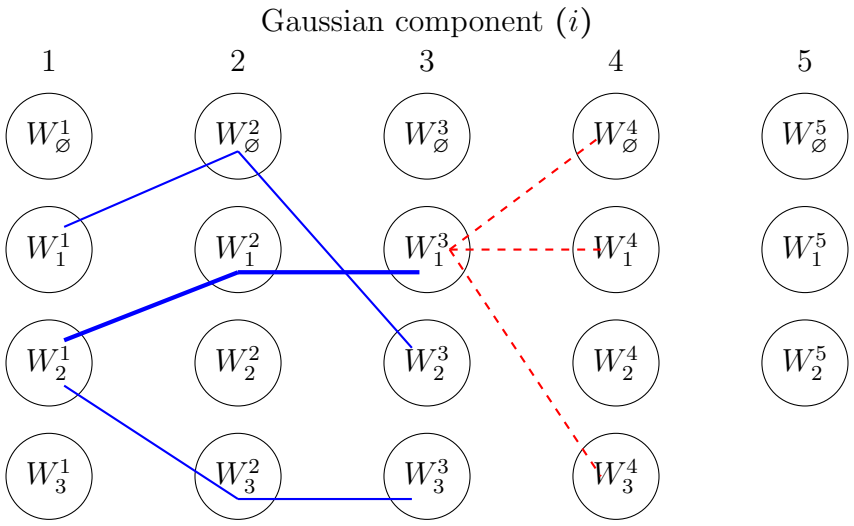


Fig. 6.2 Trellis diagram for constructing partitions. Solid blue lines represent partitions retained after processing Gaussian components 1-3. Red dashed lines correspond to partition extension when incorporating measurement subsets corresponding to the 4^{th} Gaussian component.

existing partitions are expanded using the measurement subsets from the 4^{th} component as indicated by the dashed lines. Some extensions are not included as they do not lead to a valid partition. We process the Gaussian components in decreasing order of their associated weights. After processing each component, we retain a maximum of $P_{\max}$ partitions corresponding to the ones with highest d_P. These selected partitions of measurement subsets are used in the update equations (6.47), (6.48) and (6.57) to compute the posterior PHDs and cardinality distribution.

Dancing links algorithm

For the general multisensor PHD filter a slightly more exact implementation can be used. After the first approximate step of identifying measurement subsets for each Gaussian component, instead of the approximate partition construction discussed in this section, we can find all possible partitions from the given collection of measurement subsets. This problem of finding all partitions can be mapped to the exact cover problem [160]. An efficient algorithm called Dancing Links has been suggested by Knuth [161] for solving this problem. This implementation can be used when there are relatively few sensors and the number of measurement subsets is small.

6.4 Numerical simulations

In this section we compare different multisensor multitarget tracking algorithms developed using random finite set theory. Specifically we compare the following filters - iterated-corrector PHD (IC-PHD), iterated-corrector CPHD (IC-CPHD), product multisensor PHD (P-PHD), product multisensor CPHD (P-CPHD), general multisensor PHD (G-PHD) and the general multisensor CPHD (G-CPHD) filter derived in this chapter. The models used to simulate multitarget motion and multisensor observations are discussed in Sections 6.4.1 and 6.4.2. The simulated observations are used by different algorithms to perform multitarget tracking. All the simulations were performed using algorithms implemented in Matlab running on computers with two Xeon 4-core 2.5GHz processors and 14GB RAM.

6.4.1 Target dynamics

The single target state is a four dimensional vector $\mathbf{x} = [x, y, v_x, v_y]$ consisting of its position coordinates x and y and its velocities v_x and v_y along x-axis and y-axis respectively. The target state evolves according to the discretized version of the continuous time nearly constant velocity model [158] given by

$$\mathbf{x}_{k+1,i} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{k,i} + \eta_{k+1,i} \tag{6.61}$$

$$\eta_{k+1,i} \sim \mathcal{N}(\mathbf{0}, \Sigma_\eta)\,,\ \Sigma_\eta = \begin{bmatrix} \frac{T^3}{3} & 0 & \frac{T^2}{2} & 0 \\ 0 & \frac{T^3}{3} & 0 & \frac{T^2}{2} \\ \frac{T^2}{2} & 0 & T & 0 \\ 0 & \frac{T^2}{2} & 0 & T \end{bmatrix} \sigma_\eta^2 \tag{6.62}$$

where T is the sampling period and σ_η^2 is the intensity of the process noise. We simulate 100 time steps with a sampling period of $T = 1s$ and process noise intensity of $\sigma_\eta = 0.25$.

6.4.2 Measurement model

Measurements are collected independently by multiple sensors. When a sensor detects a target, the corresponding measurement consists of the position coordinates of the target

corrupted by additive Gaussian noise. Thus if a target located at (x, y) is detected by a sensor, the measurement gathered by the sensor is given by

$$\mathbf{z} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} w_x \\ w_y \end{bmatrix} \tag{6.63}$$

where w_x and w_y are independent zero-mean Gaussian noise terms with standard deviation σ_{w_x} and σ_{w_y} respectively. In our simulations we use $\sigma_{w_x} = \sigma_{w_y} = 10$. The probability of detection of each sensor is constant throughout the monitoring region. The clutter measurements made by each of the sensors is Poisson with uniform spatial density and mean clutter rate λ.

6.4.3 Algorithm settings

For the different multisensor filters the PHD function is represented by a mixture of Gaussian densities whereas the cardinality distribution is represented by a vector of finite length which sums to one. This Gaussian mixture model approximation was first used in [41] and [90] for multitarget tracking using single sensor PHD and CPHD filters respectively. We perform pruning of Gaussian components with low weights and merging of Gaussian components in close vicinity [41] for computational tractability. For the iterated-corrector filters pruning and merging is done after processing each sensor since many components have negligible weight and propagating them has no significant impact on tracking accuracy. For the general multisensor PHD and CPHD filters pruning and merging is performed at the end of the update step since intermediate Gaussian components are not accessible. The general multisensor PHD and CPHD filters are implemented using the two-step greedy approach described in Section 6.3. In our simulations the maximum number of measurement subsets per Gaussian component is set to $W_{\max} = 6$ and the maximum number of partitions of measurement subsets is set as $P_{\max} = 6$. In later simulations we see that larger values of $W_{\max}$ and $P_{\max}$ do not provide any advantage in terms of tracking performance. For CPHD filters, the cardinality distribution is assumed to be zero for $n > 20$. This value can be much higher than the true number of targets and can be tuned using an estimate of the maximum number of targets to be tracked.

For the PHD filters, we estimate the number of targets by rounding the sum of weights of the Gaussian components to the nearest integer. For the CPHD filters, we estimate

the number of targets as the peak of the posterior cardinality distribution. For all the filters, the target state estimates are the centres of the Gaussian components with highest weights in the posterior PHD. To reduce the computational overhead, after each time step we restrict the number of Gaussian components to a maximum of four times the estimated number of targets. When the estimated number of targets is zero we retain a maximum of four Gaussian components.

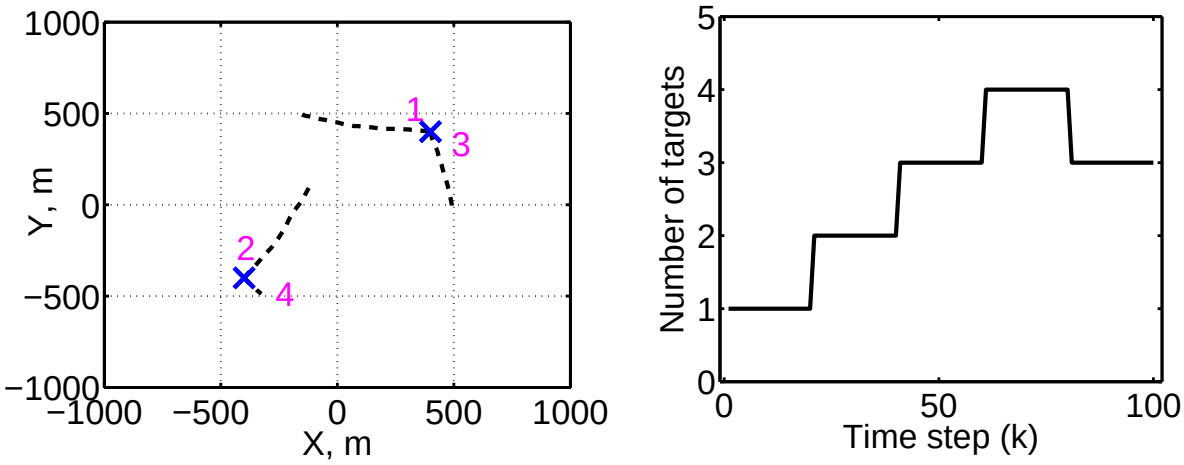
All the filters model the survival probability at all times and at all locations as constant with $p_{sv} = 0.99$. The tracking performances of the different filters are compared using the OSPA error metric discussed in Section 2.7. For the OSPA metric, we set the cardinality penalty factor $c = 100$ and power $p = 1$. These values were used in [127] for comparing different multisensor filters in a similar setup.

6.4.4 Simulation results

Experiment 1

In this experiment we generate tracks using the target motion model described in Section 6.4.1. A maximum of four targets are present at any time instant. The target tracks and the target number variation over time are shown in Figures 6.3(a) and 6.3(b) respectively. All the targets originate from one of the following two locations $(400m, 400m)$ or $(-400m, -400m)$ and target motion is restricted to the $2000m \times 2000m$ square region centered at origin. Target 1 is present in the time range $k \in [\![1, 100]\!]$; target 2 for $k \in [\![21, 100]\!]$; target 3 for $k \in [\![41, 100]\!]$; and targets 4 for $k \in [\![61, 80]\!]$. For this experiment target birth intensity is modelled as a Gaussian mixture with two components centered at $(400, 400, 0, 0)$ and $(-400, -400, 0, 0)$, each with covariance matrix $\mathrm{diag}([100, 100, 25, 25])$ and weight 0.1. In our simulations we model target birth intensity to be a Gaussian mixture density with centers at true target birth locations. In a more practical scenario where this information is unavailable the Gaussian components can be initialized based on sensor measurements, though this will result in an initial bias due to measurement noise. Target birth cardinality distribution is assumed Poisson with mean 0.2.

Three sensors gather measurements about the targets. Two of the sensors have a fixed probability of detection of 0.8. The probability of detection of the third sensor is variable and is changed from 0.2 to 1 in increments of 0.1. The mean clutter rate of each of the sensors is set to $\lambda = 10$. We consider two cases of sensor ordering where the sensor with



(a) Target tracks (b) Cardinality

Fig. 6.3 Target tracks for Experiment 1: (a) Evolution of target trajectories. (b) Number of targets as function of time.

variable probability of detection is either processed first (Case 1) or last (Case 2).

We compare all the six filters in this setup. Additionally, since the number of targets and sensors are relatively small, we also implement the general multisensor PHD filter using the Dancing Links approach (G-PHD(DL)) described in Section 6.3.3. The generated observation sequence is changed by providing a different initialization seed to the random number generator. We generate 100 different observation sequences and the average OSPA error obtained by running different multisensor filters over these 100 observation sequences is reported.

Figure 6.4(a) shows the average OSPA error as the probability of detection is changed for the two cases, Case 1 and Case 2. The approximate product filters P-PHD and P-CPHD perform significantly worse than most of the filters. This is because of their unstable nature as discussed in Section 6.1.2. For the IC-PHD filter Case 1, the accuracy improves relative to Case 2 as the probability of detection is increased since the sensor with more reliable information is processed towards the end. Figure 6.4(b) shows an enlarged portion of Figure 6.4(a). The average OSPA errors for the two different implementations of the general multisensor PHD filters are very similar. The error is slightly smaller for the G-PHD implementation compared to the G-PHD(DL) implementation because many of the non-ideal partitions get eliminated when using a greedy approach to construct partitions. The G-CPHD filter has the lowest average OSPA error.

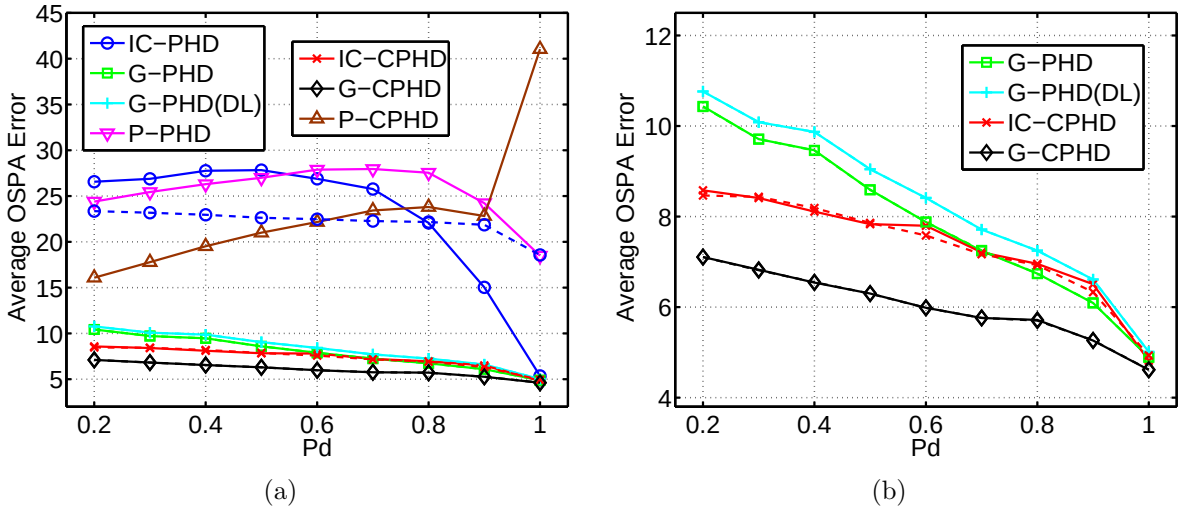


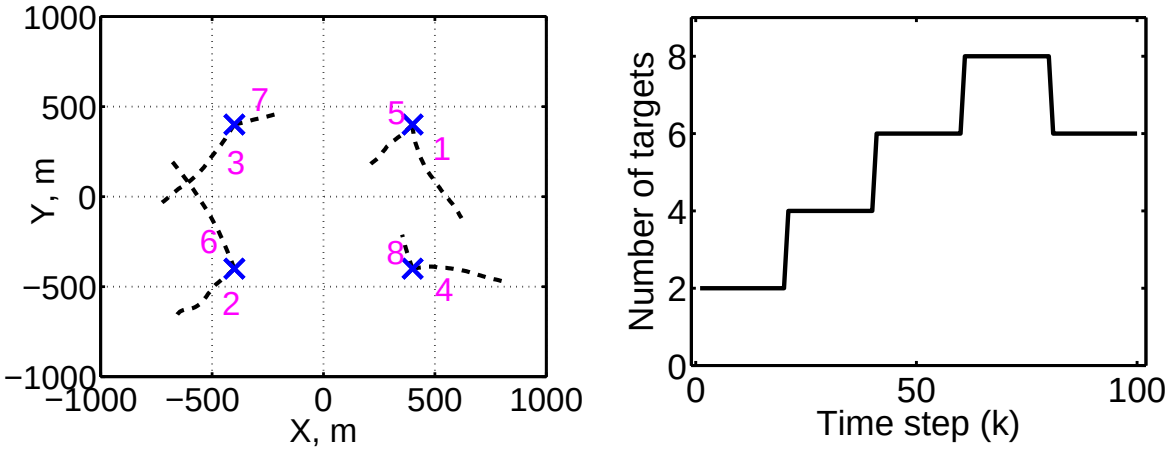
Fig. 6.4 Experiment 1: (a) Average OSPA error versus the probability of detection p_d of the variable sensor. The solid and dashed lines correspond to Case 1 and Case 2, respectively. (b) A zoomed-in version of the figure in (a) focusing on the IC-CPHD, G-PHD, G-PHD(DL) and G-CPHD filters.

Experiment 2

In this experiment we consider a maximum of eight targets being simultaneously present. Figures 6.5(a) and 6.5(b) show the target trajectories and the variation of number of targets over time. All the targets originate from one of the following four locations $(\pm 400m, \pm 400m)$ and targets are restricted to the $2000m \times 2000m$ square region centered at origin. Target birth intensity is modelled as a Gaussian mixture with four components centered at $(\pm 400, \pm 400, 0, 0)$, each with covariance matrix $\mathrm{diag}([100, 100, 25, 25])$ and weight 0.1. Target birth cardinality distribution is assumed Poisson with mean 0.4.

Six sensors gather measurements about the targets. Five of the sensors have a fixed probability of detection of 0.5. The probability of detection of the sixth sensor is variable and is changed from 0.2 to 1 in increments of 0.1. The mean clutter rate of each of the sensors is set to $\lambda = 10$. We consider two cases of sensor ordering where the sensor with variable probability of detection is either processed first (Case 1) or last (Case 2).

Since the number of sensors is large, the G-PHD(DL) filter implementation using the Dancing Links algorithm becomes infeasible (both the memory and computational requirements). Hence we only use the greedy implementation for the general multisensor PHD filter. The instability of the approximate product filters of P-PHD and P-CPHD becomes



(a) Target tracks (b) Cardinality

Fig. 6.5 Target tracks for Experiment 2 & 3: (a) Evolution of target trajectories. (b) Number of targets as function of time.

more severe when the number of sensors is large. Hence we do not include them in our discussion any further.

The average OSPA error is obtained by running different multisensor filters over 100 randomly generated observation sequences. The probability of detection of the sensor with variable probability of detection is gradually increased from 0.2 to 1. Figure 6.6(a) shows the average OSPA error as the probability of detection is changed for the two cases, Case 1 and Case 2. The IC-PHD filter performs significantly worse than all the other filters but its performance improves in Case 1 as the probability of detection is increased.

Figure 6.6(b) shows a portion of Figure 6.6(a) enlarged for clarity. We observe that for the G-PHD, IC-CPHD and G-CPHD filters there is very little difference between performance for Case 1 and Case 2. Thus the IC-CPHD filter performance does not depend significantly on the order in which sensors are processed. For the G-PHD and G-CPHD filters the order in which sensors are processed to greedily construct measurement subsets has little impact on the final filter performance. The G-CPHD filter is able to outperform both the G-PHD and the IC-CPHD filters and has the lowest average OSPA error. A box and whisker plot comparison of the G-PHD, IC-CPHD and G-CPHD filters is shown in Figure 6.7. The median OSPA error and the $25-75$ percentiles are shown for different values of p_d for the sensor with variable probability of detection.

We now examine the effect of the parameters $W_{\max}$ and $P_{\max}$, i.e., the maximum num-

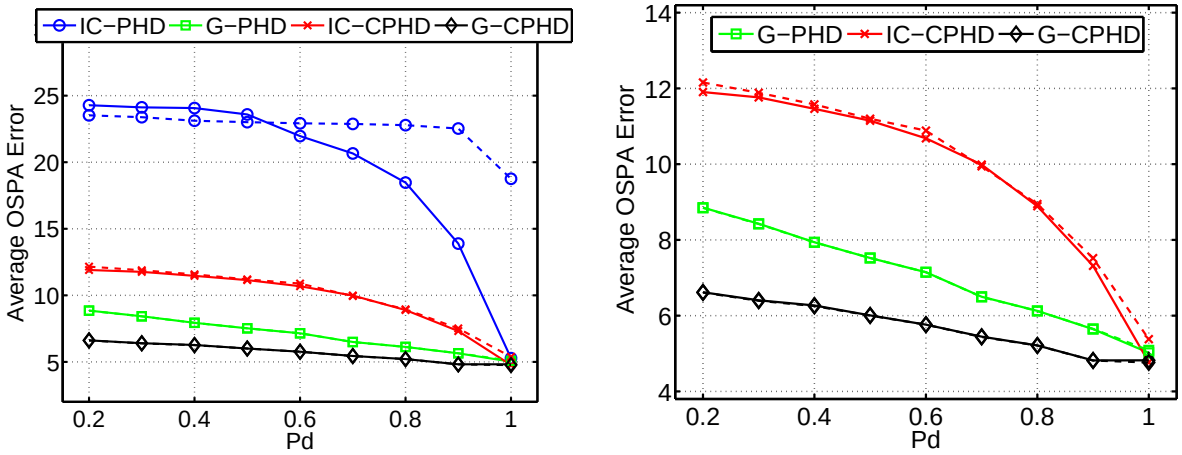


Fig. 6.6 Experiment 2: (a) Average OSPA error versus the probability of detection p_d of the variable sensor. The solid and dashed lines correspond to Case 1 and Case 2, respectively. (b) A zoomed-in version of the figure in (a) focusing on the IC-CPHD, G-PHD and G-CPHD filters.

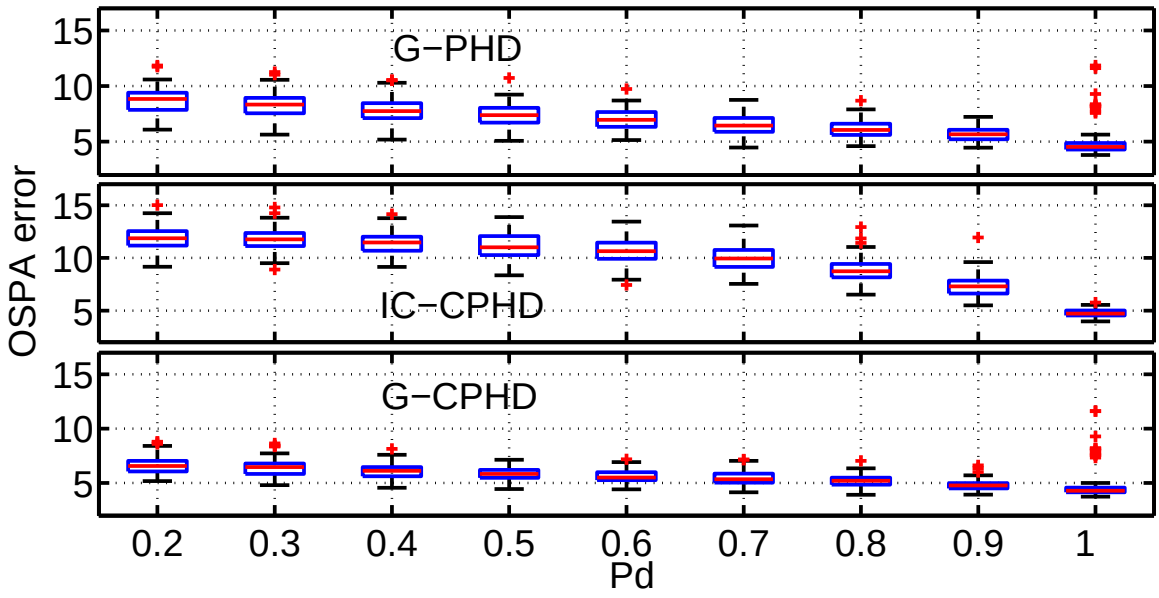


Fig. 6.7 Experiment 2: Box and whisker plot of the OSPA error as a function of p_d. Boxes indicate 25-75 interquartile range; whiskers extend 1.5 times the range and '+' symbols indicate outliers lying beyond the whiskers.

ber of measurement subsets and the maximum number of partitions using the setup of Experiment 2. $W_{\max}$ is varied in the range $\{1, 2, 4, 6, 8, 10\}$ and $P_{\max}$ is varied in the range $\{1, 2, 4, 6, 8, 10\}$. For this simulation we fix the probability of detection of all the six sensors to be 0.5. All other parameters of the simulation are the same as before. We do tracking using the same tracks as before and over 100 different observation sequences for each pair of $(W_{\max}, P_{\max})$.

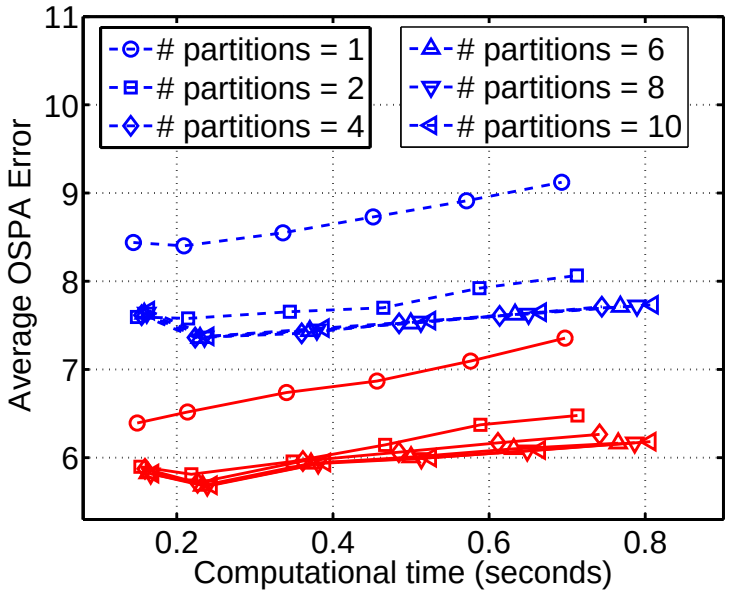


Fig. 6.8 Experiment 2: Average OSPA error Vs Computational time obtained by changing $W_{\max}$ in the range $\{1, 2, 4, 6, 8, 10\}$ and $P_{\max}$ in the range $\{1, 2, 4, 6, 8, 10\}$. Blue dashed curves correspond to G-PHD filter and red solid curves correspond to G-CPHD filter.

Figure 6.8 plots the effect of changing $W_{\max}$ and $P_{\max}$ on the average OSPA error and the average computational time required. Each curve is obtained by fixing $P_{\max}$ and changing $W_{\max}$. Dashed curves correspond to G-PHD filter and solid curves correspond to G-CPHD filters. For a given pair of $(W_{\max}, P_{\max})$ values both the filters require almost the same computational time but the G-CPHD filter has a lower average OSPA error compared to the G-PHD filter. We observe that for each curve as $W_{\max}$ increases the average OSPA error reaches a minimum quickly (around $W_{\max} = 2$) and then starts rising. This is because as $W_{\max}$ is increased the non-ideal measurement subsets also get involved in the construction of partitions leading to noise terms in the update. The computational time required grows approximately linearly with increase in $W_{\max}$. As $P_{\max}$ is varied the average OSPA error

saturates at around $P_{\max} = 4$ and increasing it beyond 4 has very little impact. Increasing $P_{\max}$ does not significantly raise the computational time requirements of the approximate G-PHD and G-CPHD filter implementations.

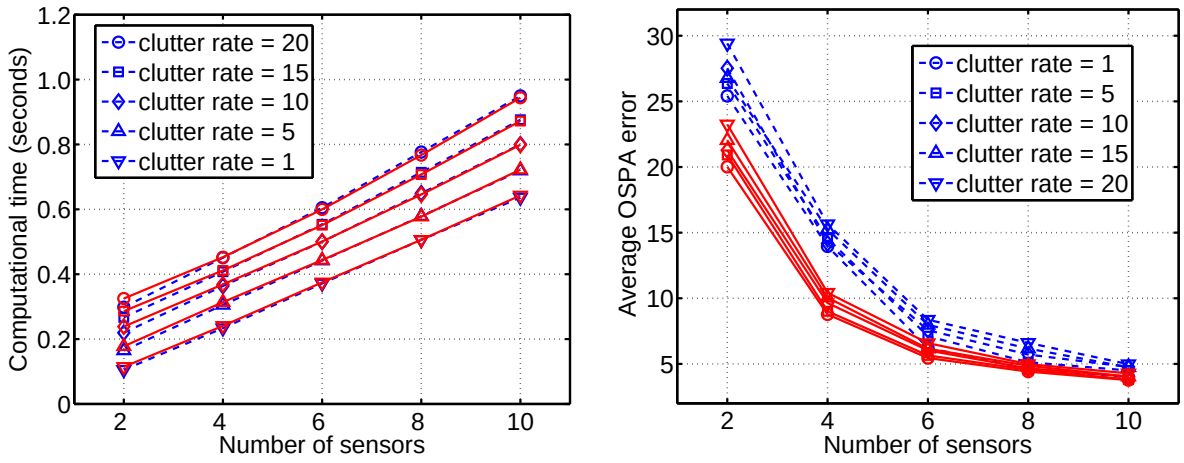
Experiment 3

To understand the scaling behavior of the approximate greedy implementations of the G-PHD and G-CPHD filters we perform another set of simulations. The target tracks described in Figures 6.5(a) and 6.5(b) are used for this experiment. We vary the number of sensors and clutter rate of the sensors in this simulation. The number of sensors is changed in the range $\{2, 4, 6, 8, 10\}$. The clutter rate is varied in the range $\{1, 5, 10, 15, 20\}$ and is the same for all the sensors. We fix the probability of detection of all the sensors to be 0.5. We set $W_{\max} = 6$ and $P_{\max} = 6$. All other parameters of the simulation are unchanged. Figures 6.9(a) and 6.9(b) plot the average computational time and the average OSPA error as the number of sensors is changed for different clutter rate values. Each curve is obtained by fixing clutter rate and changing the number of sensors. Dashed curves correspond to G-PHD filter and solid curves correspond to G-CPHD filters. From Figure 6.9(a) we observe that for approximate greedy implementations of the G-PHD and G-CPHD filters the computational requirements grow linearly with the number of sensors as well as the clutter rate.

6.4.5 Extension to non-linear measurement model

In this section we extend the Gaussian mixture based filter implementation discussed in Section 6.3 to include non-linear measurement models using the unscented Kalman filter [62, 63] approach. The unscented extensions to non-linear models when a single sensor is present are discussed in [41] and [90] for the PHD and CPHD filters respectively. We implement the unscented versions of the general multisensor PHD and CPHD filters by repeatedly applying the equations provided in [41, 90]. Specifically, the equations are recursively applied for each $\mathbf{z} \in W$ to evaluate the score function $\beta^{(i)}(W)$ while constructing the measurement subsets. Similarly, extended Kalman filter [40] based implementation of the filter proposed in Section 6.3 can also be easily derived.

As an example, we consider the setup described in [162] based on at-sea experiments. Two targets are present within the monitoring region and portions of their tracks are shown



(a) (b)

Fig. 6.9 Experiment 3: (a) Computation time required as a function of increasing number of sensors. Blue dashed curves correspond to G-PHD filter and red solid curves correspond to G-CPHD filter. (b) Average OSPA error as a function of increasing number of sensors.

in Figure 6.10(b). The target state $\mathbf{x} = [x, y]$ consists of its coordinates in the $x - y$ plane and the filters model the motion of individual targets using a random walk model given by $\mathbf{x}_{k+1,i} = \mathbf{x}_{k,i} + \eta_{k+1,i}$ where the process noise $\eta_{k+1,i}$ is zero-mean Gaussian with covariance matrix $\Sigma_\eta = \sigma_\eta^2 \,\mathrm{diag}(1, 1)$. In our simulations we set $\sigma_\eta = 0.24\,\mathrm{km}$. Although we consider linear target dynamics in this section, the unscented approach can be easily extended to include non-linear target dynamics as well.

The targets are monitored using acoustic sensors which collect bearings (angle) measurements. If sensor j is present at location $[x^j, y^j]$ and a target detected by the sensor has coordinates $[x, y]$ then the measurement made by this sensor is given by

$$z = \arctan\left(\frac{y - y^j}{x - x^j}\right) + w \tag{6.64}$$

where the measurement noise w is zero mean Gaussian with standard deviation σ_w and 'arctan' denotes the four-quadrant inverse tangent function. The sensor locations are assumed to be known. The measurements z are in the range $[0, 360)$ degrees. Along with the target related measurements the sensors also record clutter measurements not associated with any target. Five sensors (which slowly drift over time) gather measurements and their

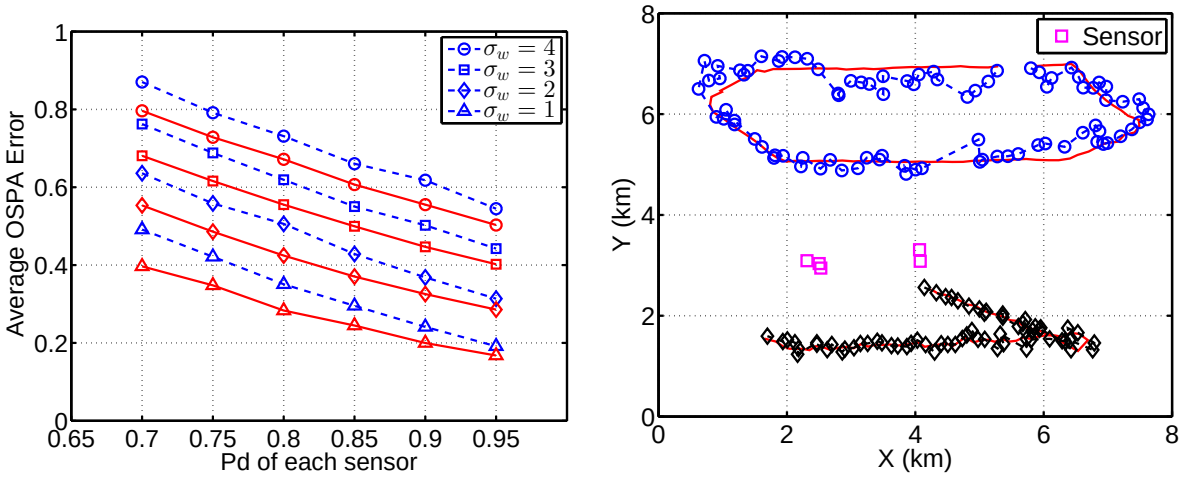
approximate locations are indicated in Figure 6.10(b). To demonstrate the feasibility of the proposed algorithms for non-linear measurement models, we consider true sensor deployments and true target trajectories with simulated measurements. We use simulated data to avoid the issue of measurement model mismatch. All sensors are assumed to have same σ_w and we vary σ_w in the range $\{1, 2, 3, 4\}$ (degrees) in our simulations. The probability of detection of each sensor is uniform throughout the monitoring region and is same for all the sensors. The probability of detection of the sensors is changed from 0.7 to 0.95 in increments of 0.05. The clutter measurements made by each of the sensors is a Poisson random finite set with uniform density in $[0, 360)$ and mean clutter rate $\lambda = 5$.

The general multisensor PHD and the general multisensor CPHD filters are used to perform tracking in this setup. Most of the implementation details are the same as discussed in Section 6.4.3. The target birth intensity is modeled as a two component Gaussian mixture with components centered at the true location of the targets at time $k = 1$ and each having covariance matrix $\text{diag}([0.65, 0.79])$ and weight 0.1. The target birth cardinality distribution is assumed to be Poisson with mean 0.2. We set $W_{\max} = 6$ and $P_{\max} = 6$. While calculating the OSPA error we use the cardinality penalty factor of $c = 2$ and power $p = 1$.

The average OSPA error obtained by running the algorithms over 100 different observation sequences are shown in Figure 6.10(a). Each curve is obtained by varying the probability of detection of the sensors from 0.7 to 0.95. As the probability of detection increases there is gradual decrease in the average OSPA error. Different curves correspond to different values of measurement noise standard deviation σ_w. As σ_w is increased the average OSPA error increases as expected. For each σ_w the G-CPHD filter performs better than the G-PHD filter. Estimated target locations obtained by the general multisensor CPHD filter are shown in Figure 6.10(b) when $\sigma_w = 2$ and $p_d = 0.9$. The tracks are obtained by joining the closest estimates across time.

6.5 Conclusions

In this chapter we addressed the problem of multitarget tracking using multiple standard sensors. Many of the existing approaches do not make complete use of the multisensor information or are computationally infeasible. As our first contribution we derived update equations for the general multisensor CPHD filter. These update equations, similar to the general multisensor PHD filter update equations, are combinatorial in nature and hence



(a) Solid lines: G-CPHD; Dashed lines: G-PHD

(b) Solid lines: True tracks; Dashed lines: Estimated tracks

Fig. 6.10 Non-linear measurement: (a): Average OSPA error Vs porbability of detection of individual sensors. Different plots obtained by changing σ_w in the range $\{1, 2, 3, 4\}$. Blue dashed curves correspond to G-PHD filter and red solid curves correspond to G-CPHD filter. (b): True target tracks (solid red) and estimated target tracks (dashed with markers) obtained using the G-CPHD filter when $\sigma_w = 2$ and $p_d = 0.9$.

computationally intractable. Our second contribution is in developing an approximate greedy implementation of the general multisensor CPHD and PHD filters based on Gaussian mixture model. The algorithm avoids any combinatorial calculations without sacrificing tracking accuracy. The algorithm is also scalable since the computational requirements grow linearly in the number of sensors as observed from the simulations.

Chapter 7

Conclusions and future work

In this chapter we provide concluding remarks for the thesis and discuss opportunities for future research and enhancements.

We motivated the problem of multitarget multisensor tracking using the radio frequency tomography application. Identifying the limitations of traditional filtering methods, we migrated towards the random finite set framework for multitarget multisensor tracking. We derived and implemented several approximations to the optimal Bayes filter within the random finite set framework.

The superpositional sensor model generalizes the radio frequency tomography measurement model. We developed particle filter implementations of the approximate PHD and CPHD filters for superpositional sensors. We also derived multi-Bernoulli filter and hybrid multi-Bernoulli CPDH filter for superpositional sensors and their particle filter implementations. Numerical studies were conducted to compare performances of these filters using simulated radio frequency tomography and simulated acoustic sensor network observations.

Finally, we derived a general multisensor CPHD filter for standard sensors. A greedy algorithm based on Gaussian mixture model-based approximation was proposed for computational tractability. Numerical studies were conducted to compare the proposed filter with other multisensor filters.

The work developed in this thesis can be extended in multiple directions, both theoretical and practical. In the following sections we discuss some of the limitations of our research and provide directions for future exploration.

7.1 Radio frequency tomography

We proposed empirical measurement models for outdoor and indoor deployment of radio frequency sensor networks. The observed heavy tails in the measurement noise are not accurately captured by the Gaussian approximations we use, but are sufficient for tracking since the measurement dimension is high. To reduce model mismatch and improve tracking accuracy, more accurate likelihood models can be explored in future work.

Through-the-wall tracking was demonstrated but was limited to the case when a single target is present and required relatively dense deployment of the sensors. Future research can focus on developing algorithms with high robustness to noise that can perform multi-target tracking with fewer sensors.

The current likelihood model assumes prior knowledge of the sensor locations. In challenging practical applications such as search-and-rescue operations this information may not be available. Integrating a sensor localization system within the multitarget tracker can solve this problem.

Although we demonstrated tracking of a time-varying number of targets, the estimate of the number of targets was poor. It would be interesting to apply the more advanced filters such as the PHD, CPHD, multi-Bernoulli and hybrid multi-Bernoulli CPHD filters developed in Chapters 4 and 5 for tracking an unknown number of targets using field data.

7.2 Multitarget tracking using superpositional sensors

The approximation techniques developed in this thesis for deriving multi-Bernoulli filters for superpositional sensors can be used for deriving other RFS based filters such as the recent labelled RFS based filters [88,89,147]. We made Gaussian approximations and applied Campbell's theorem to derive computationally tractable update equations for superpositional sensors. The analysis in Appendix A shows that this approximation is reasonable but not entirely accurate. A better solution would be to use a Gaussian mixture approximation. Finding parameters for the different Gaussian components remains a challenge as Campbell's theorem cannot be directly applied when multiple Gaussian components are present.

We developed auxiliary particle filter based implementations of various filters for superpositional sensors. We performed extensive numerical studies of these filters but they were

limited to simulated radio frequency tomography and simulated acoustic sensor network setups. Data collected from radio frequency sensor networks in Chapter 3 can be processed using these algorithms for tracking a time varying number of targets. It would be interesting to study the performance of these filters in other applications where measurements have superpositional form. Some example applications are direction-of-arrival estimation in antenna arrays and multi-path channel state estimation for MIMO-OFDM communication systems. In the problem of multi-path channel state estimation the noise is often modelled as complex random variable and the approximate update equations we develop in this thesis should be modified to account for this.

In many practical applications all the sensor measurements may not be available at a central computing entity, but rather individual sensors can process measurements locally and communicate with each other. Developing distributed algorithms to process superpositional sensor observations is a challenging problem of practical significance.

7.3 Multitarget tracking using standard sensors

We developed a greedy implementation of the general multisensor CPHD filter based on a Gaussian mixture model representation for the PHD. Since many of the multisensor filters have a combinatorial computational complexity, the greedy method for construction of measurement subsets and partitions developed in this thesis can be potentially used in other filters to reduce their computational requirements.

We extended the greedy implementation for bearings-only tracking using the unscented Kalman filter approach. Though this leads to a computationally tractable filter, it is expected to perform poorly if the measurement model is highly non-linear. Developing a computationally tractable particle filter based implementation, which can handle highly non-linear models, will be a challenging problem to study.

Our numerical study of the general multisensor CPHD filter is limited to simulated data. Future work can analyse the behavior of this filter when there is mismatch between the data and the assumed measurement models. For example, it was observed that true measurement data from sea-trial experiments violate one of the important assumptions of the filter that each target can generate at most one measurement at each sensor. This is because of the extended nature of targets (ships) which can cause multiple sonar reflections leading to multiple detections. Generalizing the filter to process such non-standard measurements is

a challenging theoretical problem.

One limitation of the filters we develop in this thesis is that they lack the information to connect the target state estimates over time and provide trajectories of individual targets. For surveillance and military applications, identification and classification of target tracks is an important requirement. Recent work on labelled random finite sets [88, 89, 147] provides an integrated approach to propagate state and label information of targets over time. Developing labelled RFS based filters which can process multisensor information is a problem of both theoretical and practical interest.

Appendix A

A.1 Gaussian approximation for superpositional sensors

To derive computationally tractable approximate filters of PHD, CPHD,multi-Bernoulli and hybrid multi-Bernoulli CPHD for superpositional sensors, one of the key steps is to approximate the density of the predicted observation vector with a Gaussian distribution. This approximation allows us to analytically simplify the integrals in the equations (4.19), (4.48), and (5.25) and provide computational tractability. Here we analyze the error introduced in the calculation of the integral and the pseudo-likelihood because of the Gaussian approximation.

Let Ξ be a random finite set with multitarget density function $f_{\Xi}(W)$. Let the PHD and second factorial moment of the random finite set Ξ be denoted by $D(\mathbf{x})$ and $D(\{\mathbf{x}_1, \mathbf{x}_2\})$. The random vector $\mathbf{y}$ is a function of the random finite set Ξ and has the following superpositional form

$$\mathbf{y} = \zeta(\Xi) = \sum_{\mathbf{x} \in \Xi} g(\mathbf{x}). \tag{A.1}$$

We are interested in the computation of the following set integral

$$\mathbf{I} = \int_{\mathcal{X}} \mathcal{N}_{\Sigma_0}(\mathbf{y}_0 - \zeta(W))\, f_{\Xi}(W) \delta W. \tag{A.2}$$

By applying the change of variables formula for set integrals (Section 2.2.4) we have the following standard integral

$$\mathbf{I} = \int_{\mathcal{Y}} \mathcal{N}_{\Sigma_0}(\mathbf{y}_0 - \mathbf{y})\, Q(\mathbf{y}) d\mathbf{y} \tag{A.3}$$

where $Q(\mathbf{y})$ is the probability density function of the random vector $\mathbf{y}$. Even though in the above form the integral is simpler to evaluate than the set integral, without making any assumptions on the density function $Q(\mathbf{y})$ the integral is analytically intractable. To make the above integral tractable, we use the approximation that the density function $Q(\mathbf{y})$ is a Gaussian density function. The mean vector and the covariance matrix of this Gaussian density function are obtained using Campbell's theorem. If $\mathbf{m}_Q$ and Σ_Q are the mean and covariance matrix then we have the approximation $Q(\mathbf{y}) \approx \mathcal{N}_{\Sigma_Q}(\mathbf{m}_Q - \mathbf{y})$. From Campbell's theorem we have

$$\mathbf{m}_Q = E[(\mathbf{y})] = \int_{\mathcal{X}} g(\mathbf{x})\, D(\mathbf{x}) d\mathbf{x} \tag{A.4}$$

$$\Sigma_Q = E[(\mathbf{y} - \mathbf{m}_Q)(\mathbf{y} - \mathbf{m}_Q)^T] \tag{A.5}$$

$$= \int_{\mathcal{X}} g(\mathbf{x})\, g(\mathbf{x})^T\, D(\mathbf{x}) d\mathbf{x} + \int_{\mathcal{X}} \int_{\mathcal{X}} g(\mathbf{x}_1)\, g(\mathbf{x}_2)^T\, \tilde{D}(\{\mathbf{x}_1, \mathbf{x}_2\}) d\mathbf{x}_1 d\mathbf{x}_2 \tag{A.6}$$

$$\text{where } \tilde{D}(\{\mathbf{x}_1, \mathbf{x}_2\}) = D(\{\mathbf{x}_1, \mathbf{x}_2\}) - D(\mathbf{x}_1) D(\mathbf{x}_2). \tag{A.7}$$

Using this Gaussian approximation for the density function $Q(\mathbf{y})$ we have the following approximation for the integral $\mathbf{I}$

$$\mathbf{I} \approx \mathbf{I}_1 = \int_{\mathcal{Y}} \mathcal{N}_{\Sigma_0}(\mathbf{y}_0 - \mathbf{y}) \mathcal{N}_{\Sigma_Q}(\mathbf{m}_Q - \mathbf{y}) d\mathbf{y} \tag{A.8}$$

$$\mathbf{I}_1 = \mathcal{N}_{\Sigma_0 + \Sigma_Q}(\mathbf{y}_0 - \mathbf{m}_Q). \tag{A.9}$$

A.1.1 Numerical analysis of approximate integral

In this section we do numerical simulations to test the validity of the approximation $\mathbf{I} \approx \mathbf{I}_1$ where $\mathbf{I}$ and $\mathbf{I}_1$ are given in expressions (A.3) and (A.9). To numerically evaluate the integral $\mathbf{I}$ we first generate samples from the random finite set Ξ and use them to generate samples of the random vector $\mathbf{y}$ using the relation in (A.1). We use N_s sample points of the random finite set Ξ to evaluate the integral $\mathbf{I}$.

To approximate the integral $\mathbf{I}_1$ we numerically compute $\mathbf{m}_Q$ and Σ_Q from (A.4) and (A.6) by using particle approximations for $D(\mathbf{x})$ and $\tilde{D}(\{\mathbf{x}_1, \mathbf{x}_2\})$. N_v sample points are used to evaluate the integral $\mathbf{I}_1$. Note that typically we need N_s to be much larger than N_v since to efficiently sample a random finite set we need many more samples than to sample from the single target state space.

We consider a two dimensional state space $\mathbf{x} = [x; y]$ consisting of the x and y coordinates of the target. For numerical analysis we consider the following linear measurement model

$$g(\mathbf{x}) = H\mathbf{x} \tag{A.10}$$

$$H = \begin{bmatrix} 0.1 & 0.7 \\ 0.5 & 0.5 \\ 0.8 & 0.2 \end{bmatrix} \tag{A.11}$$

We consider three types of random finite sets in our numerical analysis. The IIDC random finite set, the multi-Bernoulli random finite set, and the union of an IIDC and a multi-Bernoulli random finite set.

IIDC RFS

First consider Ξ to be an IIDC random finite set. Let the IIDC random finite set be described by the following normalized PHD function

$$s(\mathbf{x}) = \frac{1}{n_0} \sum_{i=1}^{n_0} \mathcal{N}_{\Sigma_i}(\mathbf{x} - \mu_i). \tag{A.12}$$

and let the cardinality distribution $\pi(n)$ be Poisson with mean n_0. For numerical simulation purposes we truncate the Poisson at $n = 15$ and normalize the cardinality distribution so it sums to 1. The above IIDC example models the case when n_0 targets are present with each target represented by a component in the Gaussian mixture. A sample of the IIDC random finite set is obtained by first sampling the cardinality n and then sampling n elements from the density function $s(\mathbf{x})$. We generate N_s samples from the IIDC random finite set Ξ and use it to calculate samples of the random vector $\mathbf{y}$ using the relation in (A.1).

The covariance matrix for each of the components is the same and is set to $\Sigma_i = \text{diag}(1, 1), i = 1, 2, \ldots, n_0$. Also set $\Sigma_0 = 0.25\,\text{diag}(1, 1, 1)$ and $\mathbf{y}_0 = H \sum_{i=1}^{n_0} g(\mu_i)$. We vary n_0 in the range $\{3, 4, 5, 6, 7, 8\}$. For each value of n_0 we run 25 different trials with the mean of the different Gaussian components μ_i in the normalized PHD function $s(\mathbf{x})$ randomly distributed over the $20m \times 20m$ region in each trial. For each trial the numerical integrals of $\mathbf{I}$ and $\mathbf{I}_1$ are evaluated 25 times and averaged to obtain their estimates $\hat{\mathbf{I}}$ and $\hat{\mathbf{I}}_1$. For evaluation of the integrals we use $N_s = 500,000$ and $N_v = 10,000$. The percentage error for

each trial is calculated as follows

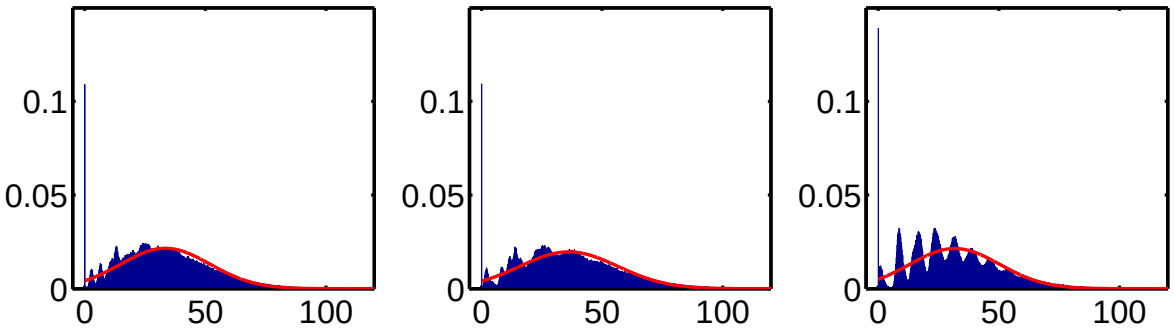
$$\text{Percentage error} = \frac{|\hat{\mathbf{I}} - \hat{\mathbf{I}}_1|}{\hat{\mathbf{I}}} \times 100. \tag{A.13}$$

To understand the error introduced due to the approximation $Q(\mathbf{y}) \approx \mathcal{N}_{\Sigma_Q}(\mathbf{m}_Q - \mathbf{y})$ we pictorially compare the normalized histograms of the elements of the random vector $\mathbf{y}$ with their Gaussian density function approximations. The histograms and the approximated Gaussian density functions are shown in Figure A.1. Histograms of each component of the vector $\mathbf{y}$ are shown for three different cases, $n_0 = 4, 6, \&8$. The histograms are plotted by dividing the data into 1000 uniform bins. $N_s = 500,000$ samples are used to generate the histograms. A Gaussian distribution function is overlaid on each histogram which has mean and variance as computed from the Campbell equations. The mean and variance are calculated using $N_v = 10,000$ sample points. As n_0 is increased we observe that the histograms better approximate the Gaussian distribution function.

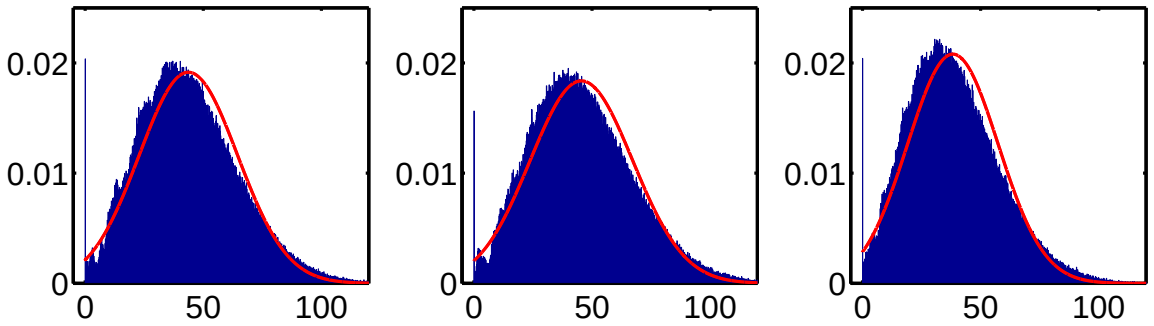
The average percentage error (computed over 25 trials) is shown in Table A.1 as the number of targets n_0 is increased. We observe a decrease in the average percentage error as the number of targets n_0 is increased. An intuitive explanation for this decrease in error and better Gaussian approximation of the histograms as n_0 is increased can be given by the application of the Central Limit Theorem to equation (A.1).

Table A.1 Average percentage error for IIDC RFS

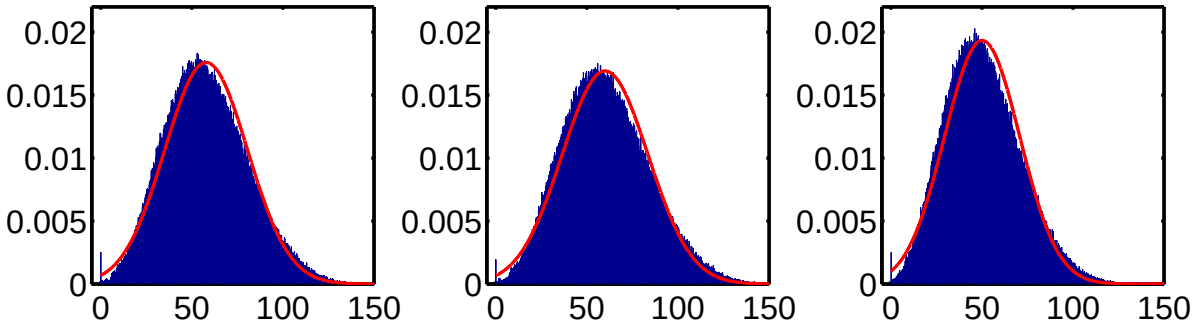
n_0	percentage error
3	66.83
4	47.63
5	27.52
6	10.72
7	4.90
8	5.29



(a) Histograms and corresponding Gaussian density approximations for each of the three components of random vector $\mathbf{y}$ ($n_0 = 4$).



(b) Histograms and corresponding Gaussian density approximations for each of the three components of random vector $\mathbf{y}$ ($n_0 = 6$).



(c) Histograms and corresponding Gaussian density approximations for each of the three components of random vector $\mathbf{y}$ ($n_0 = 8$).

Fig. A.1 Histograms (blue) and corresponding Gaussian density approximations (red) for each of the three components of random vector $\mathbf{y}$. The three rows correspond to mean cardinality $n_0 = 4, 6, \text{and } 8$. Histograms are calculated using 500,000 sample points from an IIDC random finite set.

Multi-Bernoulli RFS

Now consider Ξ to be a multi-Bernoulli random finite set. Let there be n_0 Bernoulli components with parameters

$$\{r_i, p_i(\mathbf{x})\}, i = 1, 2, \ldots n_0 \tag{A.14}$$

$$p_i(\mathbf{x}) = \mathcal{N}_{\Sigma_i}(\mathbf{x} - \mu_i) \tag{A.15}$$

The above multi-Bernoulli random finite set example represents the case when n_0 targets are present, with probability of existence r_i and each with a Gaussian density distribution. In our simulations we set $r_i \in [0.2, 1]$. Samples of the multi-Bernoulli random finite set are generated by sampling existence variable for each component and then sampling from the corresponding density function if the component exists. The mean μ_i and covariance Σ_i are same as those considered for the IIDC case.

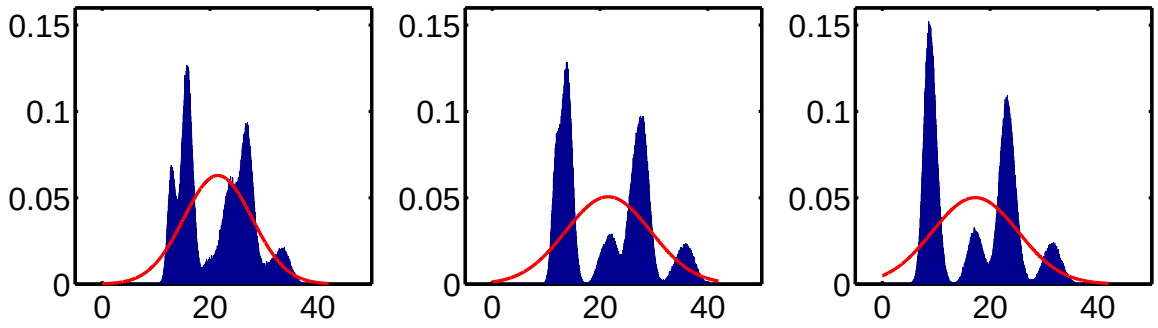
The histograms of elements of the vector $\mathbf{y}$ are shown in Figure A.2. The individual Bernoulli components have a Gaussian density function and there are 4,6 and 8 components respectively in the three sub-figures. A Gaussian function is overlaid on each histogram which has mean and variance as computed from the Campbell equations. The average percentage error (computed over 25 trials) is shown in Table A.2 as the number of targets n_0 is increased.

Table A.2 Average percentage error for multi-Bernoulli RFS

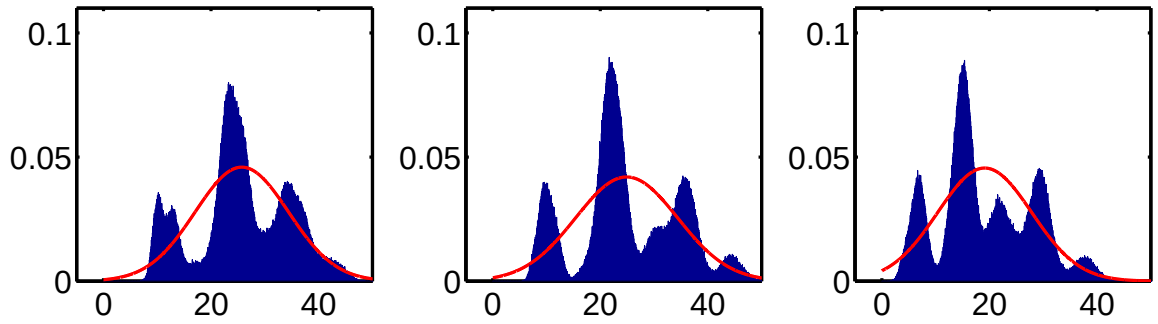
n_0	percentage error
3	76.04
4	77.24
5	76.84
6	74.73
7	71.55
8	66.84

Union of multi-Bernoulli and IIDC RFS

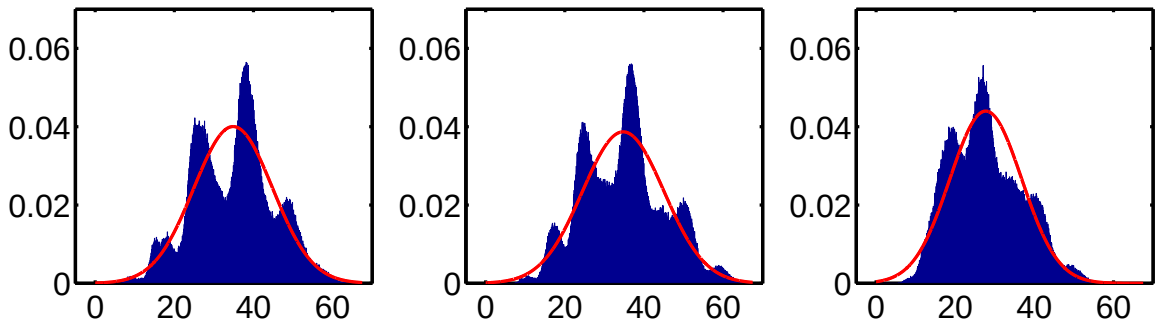
We now consider the random finite set which is union of a multi-Bernoulli RFS and an IIDC RFS. For simulations we choose the multi-Bernoulli RFS with same parameters as above. The IIDC component has a uniform discrete cardinality distribution and its normalized



(a) Histograms and corresponding Gaussian density approximations for each of the three components of random vector $\mathbf{y}$ ($n_0 = 4$).



(b) Histograms and corresponding Gaussian density approximations for each of the three components of random vector $\mathbf{y}$ ($n_0 = 6$).



(c) Histograms and corresponding Gaussian density approximations for each of the three components of random vector $\mathbf{y}$ ($n_0 = 8$).

Fig. A.2 Histograms (blue) and corresponding Gaussian density approximations (red) for each of the three components of random vector $\mathbf{y}$. The three rows correspond to $n_0 = 4, 6,$ and 8 Bernoulli components. Histograms are calculated using 500,000 sample points from a multi-Bernoulli RFS.

PHD is uniform over the $20m \times 20m$ region under consideration. The sampling process is as described earlier.

The histograms and the approximated Gaussian density functions are shown in Figure A.3. The average percentage error (computed over 25 trials) is shown in Table A.3 as the number of targets n_0 of the multi-Bernoulli component is increased.

Table A.3 Average percentage error for union of multi-Bernoulli and IIDC

n_0	percentage error
3	80.61
4	65.22
5	47.47
6	30.92
7	16.38
8	10.23

A.1.2 Numerical analysis of approximate pseudo-likelihood

Let the random finite set Ξ be union of independent random finite sets Ξ_A and Ξ_B with PHDs $D^A(\mathbf{x})$ and $D^B(\mathbf{x})$ and densities $f_{\Xi_A}(W)$ and $f_{\Xi_B}(W)$ respectively. For the case of PHD and CPHD filter analysis the component Ξ_B is empty. All of the approximate PHD update equations for superpositional sensors involve computation of the pseudo-likelihood $\mathcal{L}(\mathbf{x})$ function of the form

$$\mathcal{L}(\mathbf{x}) = \frac{\int_{\mathcal{X}} \mathcal{N}_{\Sigma_0}(\mathbf{y}_0 - g(\mathbf{x}) - \zeta(W))\, f_{\Xi}^{A^*_{\mathbf{x}}}(W)\delta W}{\int_{\mathcal{X}} \mathcal{N}_{\Sigma_0}(\mathbf{y}_0 - \zeta(W))\, f_{\Xi}(W)\delta W}, \tag{A.16}$$

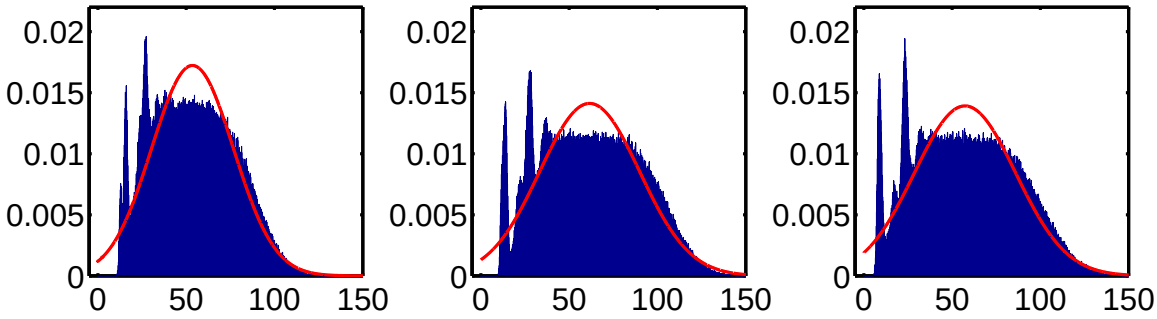
where

$$f_{\Xi}^{A^*_{\mathbf{x}}}(W) = \sum_{Y \subseteq W} \frac{f_{\Xi_A}(\{\mathbf{x}\} \cup Y)}{D^A(\mathbf{x})}\, f_{\Xi_B}(W - Y) \tag{A.17}$$

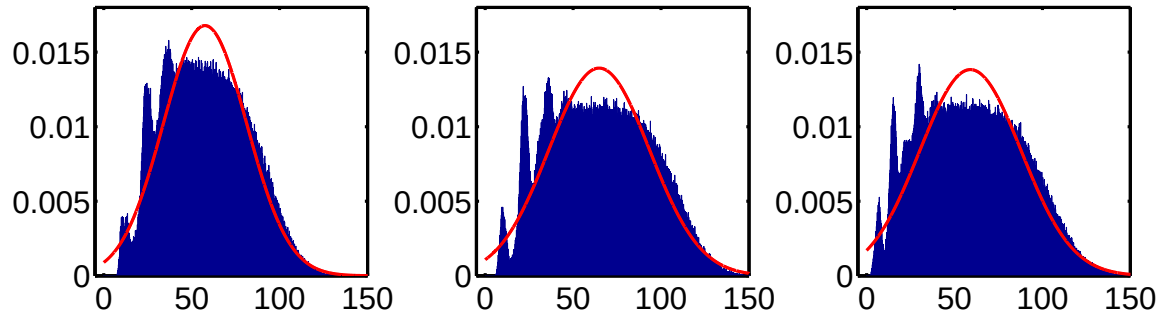
Using change of variables the ratio $\mathcal{L}(\mathbf{x})$ can be expressed as

$$\mathcal{L}(\mathbf{x}) = \frac{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_0}(\mathbf{y}_0 - g(\mathbf{x}) - \mathbf{y}^*)\, Q^{A^*_{\mathbf{x}}}(\mathbf{y}^*)d\mathbf{y}^*}{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_0}(\mathbf{y}_0 - \mathbf{y})\, Q(\mathbf{y})d\mathbf{y}} \tag{A.18}$$

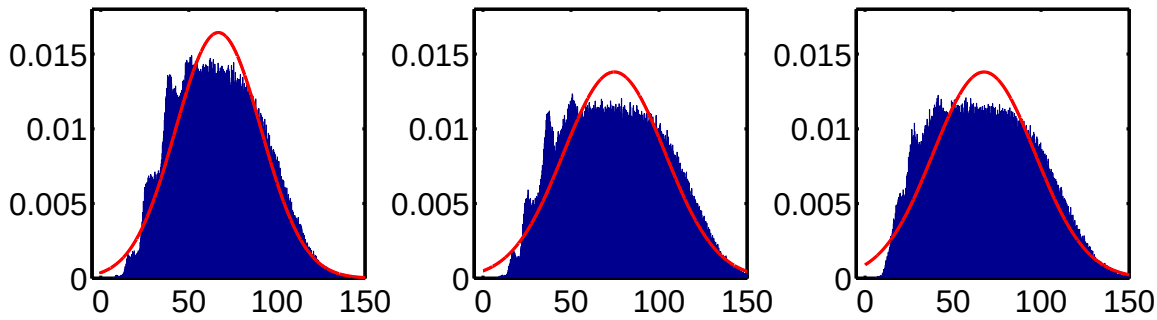
Arguing as before, to make the above ratio tractable we approximate the densities $Q(\mathbf{y})$



(a) Histograms and corresponding Gaussian density approximations for each of the three components of random vector $\mathbf{y}$ ($n_0 = 4$).



(b) Histograms and corresponding Gaussian density approximations for each of the three components of random vector $\mathbf{y}$ ($n_0 = 6$).



(c) Histograms and corresponding Gaussian density approximations for each of the three components of random vector $\mathbf{y}$ ($n_0 = 8$).

Fig. A.3 Histograms (blue) and corresponding Gaussian density approximations (red) for each of the three components of random vector $\mathbf{y}$. The three rows correspond to $n_0 = 4, 6,$ and 8 Bernoulli components. Histograms are calculated using 500,000 sample points from a union of a multi-Bernoulli and an IIDC random finite set.

and $Q^{A^*_{\mathbf{x}}}(\mathbf{y}^*)$ to be Gaussian and compute their mean and covariance matrix parameters using Campbell's theorem. Thus we have the approximation

$$\mathcal{L}(\mathbf{x}) \approx \mathcal{L}_1(\mathbf{x}) = \frac{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_0}(\mathbf{y}_0 - g(\mathbf{x}) - \mathbf{y}^*)\mathcal{N}_{\Sigma_Q^{A^*_{\mathbf{x}}}}(\mathbf{m}_Q^{A^*_{\mathbf{x}}} - \mathbf{y})d\mathbf{y}^*}{\int_{\mathcal{Y}} \mathcal{N}_{\Sigma_0}(\mathbf{y}_0 - \mathbf{y})\mathcal{N}_{\Sigma_Q}(\mathbf{m}_Q - \mathbf{y})d\mathbf{y}} \tag{A.19}$$

$$\mathcal{L}_1(\mathbf{x}) = \frac{\mathcal{N}_{\Sigma_0+\Sigma_Q^{A^*_{\mathbf{x}}}}(\mathbf{y}_0 - g(\mathbf{x}) - \mathbf{m}_Q^{A^*_{\mathbf{x}}})}{\mathcal{N}_{\Sigma_0+\Sigma_Q}(\mathbf{y}_0 - \mathbf{m}_Q)}. \tag{A.20}$$

To analyse the error introduced in the calculation of the pseudo-likelihood $\mathcal{L}(\mathbf{x})$ due the Gaussian approximation we numerically evaluate $\mathcal{L}(\mathbf{x})$ and $\mathcal{L}_1(\mathbf{x})$ for different values of $\mathbf{x}$ and compute the correlation coefficient between these two quantities. We calculate the pseudo-likelihoods for the multi-Bernoulli random finite set case when four components are present

$$\{r_i, p_i(\mathbf{x})\}, i = 1, 2, 3, 4 \tag{A.21}$$

$$p_i(\mathbf{x}) = \mathcal{N}_{\Sigma_i}(\mathbf{x} - \mu_i), i = 1, 2, 3, 4 \tag{A.22}$$

$$\Sigma_i = \text{diag}(1, 1), i = 1, 2, 3, 4. \tag{A.23}$$

We perform simulations for 100 different sets of randomly distributed means $\mu_i, i = 1, 2, 3, 4$ over the $20m \times 20m$ region. For each set of means the pseudo-likelihoods $\mathcal{L}(\mathbf{x})$ and $\mathcal{L}_1(\mathbf{x})$ are numerically computed for 100 different values of $\mathbf{x}$ sampled from a Gaussian density centered around one of the means and with covariance matrix $\text{diag}(2, 2)$. Thus the correlation coefficient is computed using $10,000$ sample points.

Example 1: The probabilities of existence of the Bernoulli components are in the range $r_i \in [0.2, 1]$. The correlation coefficient is 0.22. When the probabilities of existence are increased and are in the range $r_i \in [0.5, 1]$, the correlation coefficient increases to 0.44. A bin plot of the quantities $\mathcal{L}(\mathbf{x})$ and $\mathcal{L}_1(\mathbf{x})$ is shown in Figure A.4. The plot is generated by dividing the data into 10 groups. The x-axis points indicate the mean of each group. The red marker is the mean, the black marker is the median and the blue lines indicate 10-90 percentiles. The blue diagonal line corresponds to the case $\mathcal{L}(\mathbf{x}) = \mathcal{L}_1(\mathbf{x})$. The relatively large spread of the 10-90 percentiles is captured by the low correlation coefficient of 0.44.

Example 2: We now consider the probabilities of existence of all the components to be equal $r_i = r, i = 1, 2, 3, 4$ and increase them from $r = 0.75$ to $r = 0.99$ and compute the

correlation coefficient for each case. The correlation coefficients are given in Table A.4. For higher values of r the correlation coefficient is high indicating the approximation is more accurate for larger values of r. A bin plot of the quantities $\mathcal{L}(\mathbf{x})$ and $\mathcal{L}_1(\mathbf{x})$ when $r = 0.95$ is shown in Figure A.5. The 10-90 percentiles are narrower and the medians are closer to the diagonal line.

Table A.4 Correlation coefficients between $\mathcal{L}(\mathbf{x})$ and $\mathcal{L}_1(\mathbf{x})$

r	correlation coefficient
0.75	0.50
0.80	0.53
0.85	0.58
0.90	0.63
0.95	0.72
0.99	0.91

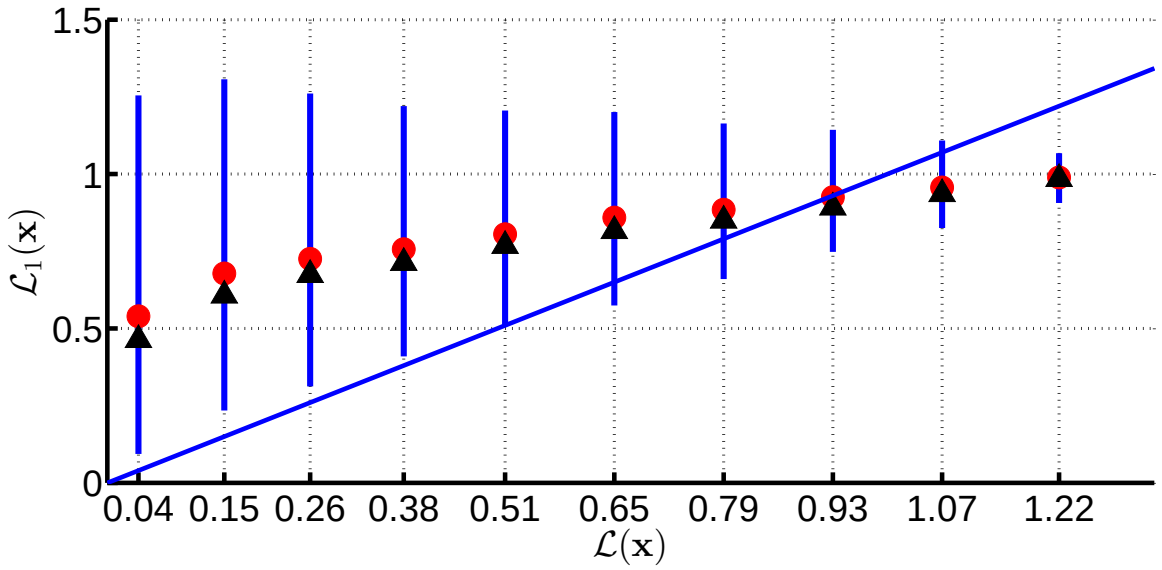


Fig. A.4 Bin plot comparing the values of $\mathcal{L}(\mathbf{x})$ and $\mathcal{L}_1(\mathbf{x})$ for the case $r_i \in [0.5, 1]$. The red marker is the mean, the black marker is the median and the vertical blue lines indicate 10-90 percentiles.

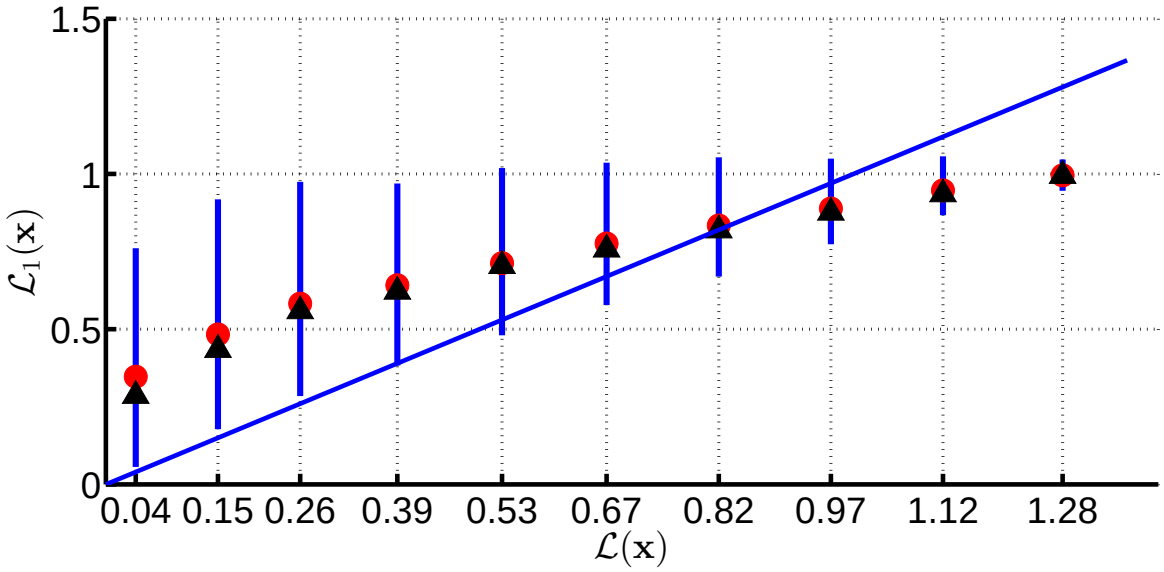


Fig. A.5 Bin plot comparing the values of $\mathcal{L}(\mathbf{x})$ and $\mathcal{L}_1(\mathbf{x})$ for the case $r = 0.95$. The red marker is the mean, the black marker is the median and the vertical blue lines indicate 10-90 percentiles.

Appendix B

B.1 Application of Campbell's theorem for the CPHD filter

Campbell's theorem was applied to derive the parameters of the approximate update equations for the CPHD filter by Mahler and El-Fallah in [38]. We present this derivation here for completeness.

The CPHD filter assumes that the predicted multitarget state is an IIDC process. Let $f_{k+1|k}(W)$ be the predicted multitarget density; $D_{k+1|k}(\mathbf{x})$ be the predicted PHD; and $\pi_{k+1|k}(n)$ be the predicted cardinality distribution at time $k+1$. Also let $s_{k+1|k}(\mathbf{x})$ be the normalized predicted PHD and $\mu_{k+1|k}$ be the mean cardinality. For this IIDC process we have

$$D_{k+1|k}(\mathbf{x}) = \mu_{k+1|k} s_{k+1|k}(\mathbf{x}), \tag{B.1}$$

$$D_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\}) = a s_{k+1|k}(\mathbf{x}_1) s_{k+1|k}(\mathbf{x}_2), \tag{B.2}$$

$$\text{where } a = \sum_{n \geq 0} n(n-1)\, \pi_{k+1|k}(n). \tag{B.3}$$

Also,

$$\tilde{D}_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\}) = D_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\}) - D_{k+1|k}(\mathbf{x}_1) D(\mathbf{x}_2) \tag{B.4}$$

$$= (a - \mu_{k+1|k}^2) s_{k+1|k}(\mathbf{x}_1) s_{k+1|k}(\mathbf{x}_2). \tag{B.5}$$

B.1.1 Parameters for PHD update

Applying Campbell's theorem we get

$$\mathbf{m}_{k+1} = \int_{\mathcal{X}} g(\mathbf{x})\, D_{k+1|k}(\mathbf{x}) d\mathbf{x} \tag{B.6}$$

$$= \mu_{k+1|k} \int_{\mathcal{X}} g(\mathbf{x})\, s_{k+1|k}(\mathbf{x}) d\mathbf{x} \tag{B.7}$$

$$= \mu_{k+1|k}\, \hat{\mathbf{m}}_{k+1}, \tag{B.8}$$

$$\text{where} \quad \hat{\mathbf{m}}_{k+1} \stackrel{\text{def}}{=} \int_{\mathcal{X}} g(\mathbf{x})\, s_{k+1|k}(\mathbf{x}) d\mathbf{x} \tag{B.9}$$

and

$$\Sigma_{k+1} = \int_{\mathcal{X}} g(\mathbf{x})\, g(\mathbf{x})^T D_{k+1|k}(\mathbf{x}) d\mathbf{x} + \int_{\mathcal{X}} \int_{\mathcal{X}} g(\mathbf{x}_1)\, g(\mathbf{x}_2)^T \tilde{D}_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\}) d\mathbf{x}_1 d\mathbf{x}_2 \tag{B.10}$$

$$= \mu_{k+1|k} \int_{\mathcal{X}} g(\mathbf{x})\, g(\mathbf{x})^T s_{k+1|k}(\mathbf{x}) d\mathbf{x} + \int_{\mathcal{X}} \int_{\mathcal{X}} g(\mathbf{x}_1)\, g(\mathbf{x}_2)^T (a - \mu^2_{k+1|k}) s_{k+1|k}(\mathbf{x}_1) s_{k+1|k}(\mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2 \tag{B.11}$$

$$= \mu_{k+1|k} \int_{\mathcal{X}} g(\mathbf{x})\, g(\mathbf{x})^T s_{k+1|k}(\mathbf{x}) d\mathbf{x} + (a - \mu^2_{k+1|k}) \left(\int_{\mathcal{X}} g(\mathbf{x}_1) s_{k+1|k}(\mathbf{x}_1) d\mathbf{x}_1 \right) \left(\int_{\mathcal{X}} g(\mathbf{x}_2)^T s_{k+1|k}(\mathbf{x}_2) d\mathbf{x}_2 \right) \tag{B.12}$$

$$= \mu_{k+1|k}\, \hat{\Sigma}_{k+1} + (a - \mu^2_{k+1|k}) \hat{\mathbf{m}}_{k+1} \hat{\mathbf{m}}^T_{k+1}, \tag{B.13}$$

$$\text{where } \hat{\Sigma}_{k+1} \stackrel{\text{def}}{=} \int_{\mathcal{X}} g(\mathbf{x})\, g(\mathbf{x})^T s_{k+1|k}(\mathbf{x}) d\mathbf{x}. \tag{B.14}$$

Now, the multitarget density $f^o_{k+1|k}(W)$ in (4.40) corresponds to an IIDC RFS with density function $s_{k+1|k}(\mathbf{x})$ and cardinality distribution $\pi^o_{k+1|k}(n)$. Its corresponding statistics

are

$$\mu^o_{k+1|k} = \sum_{n\geq 0} n\pi^o_{k+1|k}(n) \tag{B.15}$$

$$= \sum_{n\geq 0} n\frac{(n+1)\,\pi_{k+1|k}(n+1)}{\mu_{k+1|k}} \tag{B.16}$$

$$= \frac{a}{\mu_{k+1|k}}, \tag{B.17}$$

$$\text{and} \quad a^o = \sum_{n\geq 0} n(n-1)\,\pi^o_{k+1|k}(n) \tag{B.18}$$

$$= \sum_{n\geq 0} n(n-1)\,\frac{(n+1)\,\pi_{k+1|k}(n+1)}{\mu_{k+1|k}} \tag{B.19}$$

$$= \frac{b}{\mu_{k+1|k}}, \tag{B.20}$$

$$\text{where} \quad b \overset{\text{def}}{=} \sum_{n\geq 0} n(n+1)(n+2)\,\pi_{k+1|k}(n+2). \tag{B.21}$$

Thus applying Campbell's theorem we get

$$\mathbf{m}^o_{k+1} = \mu^o_{k+1|k}\,\hat{\mathbf{m}}_{k+1} \tag{B.22}$$

$$= \frac{a}{\mu_{k+1|k}}\,\hat{\mathbf{m}}_{k+1}, \tag{B.23}$$

and

$$\Sigma^o_{k+1} = \mu^o_{k+1|k}\,\hat{\Sigma}_{k+1} + (a^o - (\mu^o_{k+1|k})^2)\hat{\mathbf{m}}_{k+1}\hat{\mathbf{m}}^T_{k+1} \tag{B.24}$$

$$= \frac{a}{\mu_{k+1|k}}\,\hat{\Sigma}_{k+1} + \left(\frac{b}{\mu_{k+1|k}} - \frac{a^2}{\mu^2_{k+1|k}}\right)\hat{\mathbf{m}}_{k+1}\hat{\mathbf{m}}^T_{k+1}. \tag{B.25}$$

B.1.2 Parameters for cardinality update

From (4.63) we have

$$f^n_{k+1|k}(W) \overset{\text{def}}{=} \frac{1}{\pi_{k+1|k}(n)}\delta_{|W|,n} f_{k+1|k}(W). \tag{B.26}$$

We now prove that $f^n_{k+1|k}(W)$ is in fact a multitarget density function. Upon integration we get

$$\int_{\mathcal{X}} f^n_{k+1|k}(W)\delta W = f^n_{k+1|k}(\varnothing) + \sum_{j=1}^{\infty} \frac{1}{j!} \int_{\mathcal{X}^j} f^n_{k+1|k}(\{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_j\}) d\mathbf{w}_1 \ldots d\mathbf{w}_j \tag{B.27}$$

$$= \frac{1}{n!} \frac{1}{\pi_{k+1|k}(n)} \int_{\mathcal{X}^n} f_{k+1|k}(\{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_n\}) d\mathbf{w}_1 \ldots d\mathbf{w}_n \tag{B.28}$$

$$= \frac{1}{n!} \frac{1}{\pi_{k+1|k}(n)} \times n!\, \pi_{k+1|k}(n) \tag{B.29}$$

$$= 1. \tag{B.30}$$

The PHD of the random finite set with density function $f^n_{k+1|k}(W)$ is

$$D^n_{k+1|k}(\mathbf{x}) = \int_{\mathcal{X}} f^n_{k+1|k}(\{\mathbf{x}\} \cup W)\delta W \tag{B.31}$$

$$= \sum_{j=0}^{\infty} \frac{1}{j!} \int_{\mathcal{X}^j} f^n_{k+1|k}(\{\mathbf{x}\} \cup \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_j\}) d\mathbf{w}_1 \ldots d\mathbf{w}_j \tag{B.32}$$

$$= \frac{1}{(n-1)!} \frac{1}{\pi_{k+1|k}(n)} \int_{\mathcal{X}^{n-1}} f_{k+1|k}(\{\mathbf{x}\} \cup \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{n-1}\}) d\mathbf{w}_1 \ldots d\mathbf{w}_{n-1} \tag{B.33}$$

$$= \frac{n!\, \pi_{k+1|k}(n)}{(n-1)! \pi_{k+1|k}(n)} s_{k+1|k}(\mathbf{x}) \int_{\mathcal{X}^{n-1}} s_{k+1|k}(\mathbf{w}_1) \ldots s_{k+1|k}(\mathbf{w}_{n-1}) d\mathbf{w}_1 \ldots \mathbf{w}_{n-1} \tag{B.34}$$

$$= n\, s_{k+1|k}(\mathbf{x}). \tag{B.35}$$

The second factorial moment can be calculated as

$$D^n_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\}) = \int_{\mathcal{X}} f^n_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\} \cup W)\delta W \tag{B.36}$$

$$= \sum_{j=0}^{\infty} \frac{1}{j!} \int_{\mathcal{X}^j} f^n_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\} \cup \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_j\}) d\mathbf{w}_1 \ldots d\mathbf{w}_j \tag{B.37}$$

$$= \frac{1}{(n-2)!} \frac{1}{\pi_{k+1|k}(n)} \int_{\mathcal{X}^{n-2}} f_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\} \cup \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{n-2}\}) d\mathbf{w}_1 \ldots d\mathbf{w}_{n-2} \tag{B.38}$$

$$= \frac{n!\, \pi_{k+1|k}(n)}{(n-2)! \pi_{k+1|k}(n)} s_{k+1|k}(\mathbf{x}_1) s_{k+1|k}(\mathbf{x}_2) \times \int_{\mathcal{X}^{n-2}} s_{k+1|k}(\mathbf{w}_1) \ldots s_{k+1|k}(\mathbf{w}_{n-2}) d\mathbf{w}_1 \ldots \mathbf{w}_{n-2} \tag{B.39}$$

$$= n(n-1)\, s_{k+1|k}(\mathbf{x}_1)\, s_{k+1|k}(\mathbf{x}_2). \tag{B.40}$$

Thus

$$\tilde{D}^n_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\}) = D^n_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\}) - D^n_{k+1|k}(\mathbf{x}_1) D^n_{k+1|k}(\mathbf{x}_2) \tag{B.41}$$

$$= n(n-1)\, s_{k+1|k}(\mathbf{x}_1)\, s_{k+1|k}(\mathbf{x}_2) - n\, s_{k+1|k}(\mathbf{x}_1)\, n\, s_{k+1|k}(\mathbf{x}_2) \tag{B.42}$$

$$= -n\, s_{k+1|k}(\mathbf{x}_1)\, s_{k+1|k}(\mathbf{x}_2). \tag{B.43}$$

Applying Campbell's theorem, the parameters $\mathbf{m}^n_{k+1}$ and Σ^n_{k+1} for the Gaussian approximation in (4.65) are given by

$$\mathbf{m}^n_{k+1} = \int_{\mathcal{X}} g(\mathbf{x}) D^n_{k+1|k}(\mathbf{x}) \tag{B.44}$$

$$= \int_{\mathcal{X}} g(\mathbf{x}) n\, s_{k+1|k}(\mathbf{x}) \tag{B.45}$$

$$= n\, \hat{\mathbf{m}}_{k+1}\,, \tag{B.46}$$

and

$$\Sigma^n_{k+1} = \int_{\mathcal{X}} g(\mathbf{x})\, g(\mathbf{x})^T D^n_{k+1|k}(\mathbf{x}) d\mathbf{x} + \int_{\mathcal{X}} \int_{\mathcal{X}} g(\mathbf{x}_1)\, g(\mathbf{x}_2)^T \tilde{D}^n_{k+1|k}(\{\mathbf{x}_1, \mathbf{x}_2\}) d\mathbf{x}_1 d\mathbf{x}_2 \tag{B.47}$$

$$= n \int_{\mathcal{X}} g(\mathbf{x})\, g(\mathbf{x})^T s_{k+1|k}(\mathbf{x}) d\mathbf{x} - n \int_{\mathcal{X}} \int_{\mathcal{X}} g(\mathbf{x}_1)\, g(\mathbf{x}_2)^T s_{k+1|k}(\mathbf{x}_1)\, s_{k+1|k}(\mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2 \tag{B.48}$$

$$= n\, (\hat{\Sigma}_{k+1} - \hat{\mathbf{m}}_{k+1} \hat{\mathbf{m}}^T_{k+1}). \tag{B.49}$$

Appendix C

This appendix provides background material and intermediate results required to prove Theorem 1 of Chapter 6. A recursive expression for constructing the collection of all possible partitions $\mathcal{P}$ is given in Appendix C.1. The concepts of functional derivatives and integral transform of the posterior multitarget density are revised in Appendices C.2 and C.3, respectively. The proof of Theorem 1 depends on Lemma 1 which is proved in Appendix C.4. Finally, combining these results, the proof of Theorem 1 is provided in Appendix C.5.

C.1 Recursive expression for collection of partitions

Let $\mathcal{P}^{(\ell)}$ be the collection of all possible partitions of the set $Z_{k+1}^{1:\ell}$ $(1 \leq \ell < s)$ where partitions are as defined in the equations (6.35)-(6.38). Since the V component of a partition is unique given the W components, we do not explicitly specify the V component in the recursive expression. Let $P \in \mathcal{P}^{(\ell)}$ be any partition of $Z_{k+1}^{1:\ell}$ which is given as $P = \{W_1, W_2, \ldots, , W_{|P|-1}, V\}$. Let $Z_{k+1}^{\ell+1} = \{\mathbf{z}_1^{\ell+1}, \mathbf{z}_2^{\ell+1}, \ldots, \mathbf{z}_{m_{\ell+1}}^{\ell+1}\}$. Then we can express $\mathcal{P}^{(\ell+1)}$ using $\mathcal{P}^{(\ell)}$ and $Z_{k+1}^{\ell+1}$ as given by the following relation

$$\mathcal{P}^{(\ell+1)} = \bigcup_{P \in \mathcal{P}^{(\ell)}} \bigcup_{n_1=0}^{m_{\ell+1}} \bigcup_{n_2=0}^{\min(m_{\ell+1},|P|-1)} \bigcup_{\substack{I_1 \subseteq [\![1,m_{\ell+1}]\!] \\ |I_1|=n_1}} \bigcup_{\substack{I_2 \subseteq [\![1,m_{\ell+1}]\!] \\ |I_2|=n_2 \\ I_1 \cap I_2 = \varnothing}} \bigcup_{\substack{J \subseteq [\![1,|P|-1]\!] \\ |J|=|I_2|}} \bigcup_{B \in \mathcal{B}(I_2,J)}$$
$$\left\{ \left\{ \mathbf{z}_{i_1}^{\ell+1} \right\}_{i_1 \in I_1} \cup \left\{ W_{B(i_2)}, \mathbf{z}_{i_2}^{\ell+1} \right\}_{i_2 \in I_2} \cup \left\{ W_j \right\}_{j \notin J} \right\} \tag{C.1}$$

where $\mathcal{B}(I_2, J)$ is the collection of all possible matchings [1] from set I_2 to set J. The above relation mathematically expresses the fact that for each partition $P \in \mathcal{P}^{(\ell)}$ and given $Z_{k+1}^{\ell+1}$, a new partition belonging to $\mathcal{P}^{(\ell+1)}$ can be constructed by adding some new singleton measurement subsets from $Z_{k+1}^{\ell+1}$ (i.e. $\{\mathbf{z}_{i_1}^{\ell+1}\}_{i_1 \in I_1}$), extending some existing subsets in P by appending them with measurements from $Z_{k+1}^{\ell+1}$ (i.e. $\{W_{B(i_2)} \cup \mathbf{z}_{i_2}^{\ell+1}\}_{i_2 \in I_2}$) and retaining some existing measurement subsets (i.e. $\{W_j\}_{j \notin J}$). By this definition we have $\mathcal{P} = \mathcal{P}^{(s)}$. As a special case for $\ell = 1$,

$$\mathcal{P}^{(1)} = \bigcup_{n=0}^{m_1} \bigcup_{\substack{I \subseteq [\![1, m_1]\!] \\ |I| = n}} \left\{ \{\mathbf{z}_i^1\}_{i \in I} \right\}. \tag{C.2}$$

C.2 Functional derivatives

We now review the notion of functional derivatives which play an important role in the derivation of filter update equations. A brief background is provided that is necessary for the derivations; for additional details see [87], [34, Ch. 11]. Let $\mathcal{F}$ denote the set of mappings from $\mathcal{Y}$ to $\mathbb{R}$ and let A be a functional mapping elements of $\mathcal{F}$ to $\mathbb{R}$. Let $u(\mathbf{y})$ and $g(\mathbf{y})$ be functions in $\mathcal{F}$. For the functional $A[u]$, its functional derivative along the direction of the function $g(\mathbf{y})$ is defined as [34]

$$\frac{\partial A}{\partial g}[u] \stackrel{\text{def}}{=} \lim_{\epsilon \to 0} \frac{A[u + \epsilon \cdot g] - A[u]}{\epsilon}. \tag{C.3}$$

We are specifically interested in the functional derivatives when the function $g(\mathbf{y})$ is the Dirac delta function $\delta_{\mathbf{y}_1}(\mathbf{y})$ localized at $\mathbf{y}_1$. In such case we use the following notation for functional derivatives [34]

$$\frac{\partial A}{\partial \delta_{\mathbf{y}_1}}[u] \equiv \frac{\delta A}{\delta \mathbf{y}_1}[u]. \tag{C.4}$$

If the functional $A[u]$ is of the form $A[u] = \int u(\mathbf{y}) g(\mathbf{y}) d\mathbf{y}$ then we have

$$\frac{\delta A}{\delta \mathbf{y}_1}[u] = g(\mathbf{y}_1). \tag{C.5}$$

[1] $\mathcal{B}(I, J)$ is the collection of all possible one-to-one mappings from set I to set J.

We can also define higher order functional derivatives of $A[u]$. For a set $Y = \{\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n\}$, the n^{th} order derivative is denoted by

$$\frac{\delta^n A}{\delta Y}[u] \stackrel{\text{def}}{=} \frac{\delta^n A}{\delta \mathbf{y}_1 \delta \mathbf{y}_2 \ldots \delta \mathbf{y}_n}[u]. \tag{C.6}$$

We call $\frac{\delta^n A}{\delta Y}[u]$ the functional derivative of $A[u]$ with respect to the set Y.

Product rule

For functionals $A_1[u]$, $A_2[u]$ and $A_3[u]$, the product rule for functional derivatives [34, Ch. 11] gives

$$\frac{\delta}{\delta Y}\{A_1[u]\, A_2[u]\, A_3[u]\} = \sum_{Y_1 \subseteq Y} \sum_{\substack{Y_2 \subseteq Y \\ Y_1 \cap Y_2 = \varnothing}} \frac{\delta A_1}{\delta Y_1}[u] \frac{\delta A_2}{\delta Y_2}[u] \frac{\delta A_3}{\delta (Y - Y_1 - Y_2)}[u] \tag{C.7}$$

$$= \sum_{n_1=0}^{|Y|} \sum_{n_2=0}^{|Y|} \sum_{\substack{Y_1 \subseteq Y \\ |Y_1| = n_1}} \sum_{\substack{Y_2 \subseteq Y \\ |Y_2| = n_2 \\ Y_1 \cap Y_2 = \varnothing}} \frac{\delta A_1}{\delta Y_1}[u] \frac{\delta A_2}{\delta Y_2}[u] \frac{\delta A_3}{\delta (Y - Y_1 - Y_2)}[u]. \tag{C.8}$$

As a special case, for the product of two functionals we have

$$\frac{\delta}{\delta Y}\{A_1[u]\, A_2[u]\} = \sum_{n=0}^{|Y|} \sum_{\substack{Y_1 \subseteq Y \\ |Y_1| = n}} \frac{\delta A_1}{\delta Y_1}[u] \frac{\delta A_2}{\delta (Y - Y_1)}[u]. \tag{C.9}$$

Chain rule

Let $f(y)$ be a function mapping elements of $\mathbb{R}$ to $\mathbb{R}$, then from the chain rule [34, Ch. 11] we have

$$\frac{\delta}{\delta \mathbf{y}} f(A[u]) = \frac{df}{dy}(A[u]) \frac{\delta A}{\delta \mathbf{y}}[u]. \tag{C.10}$$

PHD as functional derivative

Recall that for a random variable its moments are related to the derivatives of its PGF. Similarly, the first moment or the PHD function of a random finite set is related to the functional derivative of its PGFL. For the random finite set Ξ, its PHD function $D_{\Xi}(\mathbf{x})$ is

related [34] to the functional derivative of the PGFL $G_{\Xi}[u]$ as follows

$$D_{\Xi}(\mathbf{x}) = \frac{\delta G_{\Xi}}{\delta \mathbf{x}}[1]. \tag{C.11}$$

C.3 Integral transform of the posterior multitarget density

Let $f_{k+1|k}(X|Z_{1:k}^{1:s})$ and $f_{k+1|k+1}(X|Z_{1:k+1}^{1:s})$ be the predicted and posterior multitarget state distributions at time $k+1$ and let $L_{k+1,j}(Z_{k+1}^{j}|X)$ denote the multitarget likelihood function for the j^{th} sensor at time $k+1$. Since the sensor observations are independent conditional on the multitarget state, the update equation for the multitarget Bayes filter [34] is given by

$$f_{k+1|k+1}(X|Z_{1:k+1}^{1:s}) \propto f_{k+1|k}(X|Z_{1:k}^{1:s}) \prod_{j=1}^{s} L_{k+1,j}(Z_{k+1}^{j}|X). \tag{C.12}$$

We now define a multivariate functional which is the integral transform of the quantity in the right hand side of the above equation. Under the conditions of Assumption 2 (Section 6.1), we can obtain a closed form expression for this multivariate functional, which on differentiation gives the PGFL of the posterior multitarget state distribution.

Let $g_j(\mathbf{z}), j = 1, 2, \ldots, s$ be functions that map the space $\mathcal{Z}^j$ to $[0,1]$ where $\mathcal{Z}^j$ is the space of observations of sensor j. The intermediate functions $g_j(\mathbf{z})$ will be used to define functionals and later set to zero to obtain the PGFL of the posterior multitarget distribution. Let $u(\mathbf{x})$ be a function mapping the state space $\mathcal{X}$ to $[0,1]$. For brevity, denote the vector of functions $[g_1, g_2, \ldots, g_s]$ as $g_{1:s}$ and define $g_j^{Z^j} \stackrel{\text{def}}{=} \prod_{\mathbf{z}\in Z^j} g_j(\mathbf{z})$ where $Z^j \subseteq \mathcal{Z}^j$. We define the multivariate functional $F[g_1, g_2, \ldots, g_s, u]$ as the following integral transform

$$F[g_{1:s}, u] \stackrel{\text{def}}{=} \int_{\mathcal{X}} u^X \left(\prod_{j=1}^{s} \mathcal{L}_{k+1,j}[g_j|X] \right) f_{k+1|k}(X|Z_{1:k}^{1:s})\, \delta X, \tag{C.13}$$

$$\text{where } \mathcal{L}_{k+1,j}[g_j|X] \stackrel{\text{def}}{=} \int_{\mathcal{Z}^j} g_j^{Z^j}\, L_{k+1,j}(Z^j|X) \delta Z^j. \tag{C.14}$$

Later we will relate the PGFL of the posterior multitarget distribution to the derivatives of the functional $F[g_{1:s}, u]$ with respect to the sensor observations $Z_{k+1}^{1:s}$. Recall that $c_j(\mathbf{z})$ denotes the clutter spatial distribution and $C_j(t)$ denotes the PGF of the clutter cardinality

distribution for the j^{th} sensor. Under Assumption 2 (Section 6.1) it can be shown that [86]

$$\mathcal{L}_{k+1,j}[g_j|X] = \int_{\mathcal{Z}^j} g_j^{Z^j} L_{k+1,j}(Z^j|X)\delta Z^j \tag{C.15}$$

$$= C_j(\langle c_j, g_j\rangle)\,\phi_{g_j}^X\,, \tag{C.16}$$

$$\text{where } \phi_{g_j}(\mathbf{x}) \stackrel{\text{def}}{=} 1 - p_d^j(\mathbf{x}) + p_d^j(\mathbf{x})\,p_{g_j}(\mathbf{x}) \tag{C.17}$$

$$p_{g_j}(\mathbf{x}) \stackrel{\text{def}}{=} \int_{\mathcal{Z}^j} g_j(\mathbf{z})h_j(\mathbf{z}|\mathbf{x})d\mathbf{z}. \tag{C.18}$$

Let $G_{k+1|k}[u]$ denote the PGFL of the predicted multitarget distribution. Using the above relations in (C.13) we have

$$F[g_{1:s}, u] = \int_{\mathcal{X}} u^X \left(\prod_{j=1}^{s} C_j(\langle c_j, g_j\rangle)\,\phi_{g_j}^X\right) f_{k+1|k}(X|Z_{1:k}^{1:s})\,\delta X \tag{C.19}$$

Since both u and ϕ_{g_j} are functions defined over the space $\mathcal{X}$, we can combine the product of u^X and $\prod_{j=1}^{s}\phi_{g_j}^X$ and write $(u\prod_{j=1}^{s}\phi_{g_j})^X$. Hence we have

$$F[g_{1:s}, u] = \left(\prod_{j=1}^{s} C_j(\langle c_j, g_j\rangle)\right) \int_{\mathcal{X}} \left(u\prod_{j=1}^{s}\phi_{g_j}\right)^X f_{k+1|k}(X|Z_{1:k}^{1:s})\,\delta X \tag{C.20}$$

$$= \left(\prod_{j=1}^{s} C_j(\langle c_j, g_j\rangle)\right) G[u\prod_{j=1}^{s}\phi_{g_j}] \tag{C.21}$$

$$= \left(\prod_{j=1}^{s} C_j(\langle c_j, g_j\rangle)\right) M(\langle r, u\prod_{j=1}^{s}\phi_{g_j}\rangle). \tag{C.22}$$

The last two steps result from the definition of the PGFL and the assumption that the predicted multitarget distribution $f_{k+1|k}(X|Z_{1:k}^{1:s})$ is IIDC.

Let $G_{k+1|k+1}[u]$ be the PGFL of the multitarget density $f_{k+1|k+1}(X|Z_{1:k+1}^{1:s})$, and let $D_{k+1|k+1}(\mathbf{x})$ be the posterior PHD function. From [19, 122] we have the following relation

$$G_{k+1|k+1}[u] = \frac{\dfrac{\delta F}{\delta Z_{k+1}^{1:s}}[0, 0, \ldots, 0, u]}{\dfrac{\delta F}{\delta Z_{k+1}^{1:s}}[0, 0, \ldots, 0, 1]}. \tag{C.23}$$

Since the PHD is the functional derivative of the PGFL, from (C.11)

$$D_{k+1|k+1}(\mathbf{x}) = \frac{\dfrac{\delta F}{\delta \mathbf{x}\, \delta Z_{k+1}^{1:s}}[0, 0, \ldots, 0, 1]}{\dfrac{\delta F}{\delta Z_{k+1}^{1:s}}[0, 0, \ldots, 0, 1]}. \tag{C.24}$$

Note that the differentiation $\frac{\delta}{\delta Z_{k+1}^{j}}$ is with respect to the function variable g_j and the differentiation $\frac{\delta}{\delta \mathbf{x}}$ is with respect to the function variable u. The general multisensor CPHD filter update equation is derived by evaluating the functional derivatives of $F[g_{1:s}, u]$ in (C.23) and (C.24).

We now define a quantity Γ and the functionals $\Psi_P[g_{1:s}, u]$ and $\varphi_W[g_{1:s}, u]$. The functional derivatives of $F[g_{1:s}, u]$ can be expressed in terms of these quantities. Let

$$\Gamma \stackrel{\text{def}}{=} \prod_{j=1}^{s} \left(\prod_{\mathbf{z} \in Z_{k+1}^{j}} c_j(\mathbf{z}) \right), \tag{C.25}$$

$$\Psi_P[g_{1:s}, u] \stackrel{\text{def}}{=} \left(\prod_{j=1}^{s} C_j^{(m_j - |P|_j)}(\langle c_j, g_j \rangle) \right) M^{(|P|-1)}(\langle r, u \prod_{j=1}^{s} \phi_{g_j} \rangle). \tag{C.26}$$

For $W \in \mathcal{W}$, let

$$\varphi_W[g_{1:s}, u] \stackrel{\text{def}}{=} \frac{\displaystyle\int_{\mathcal{X}} r(\mathbf{x}) u(\mathbf{x}) \left(\prod_{(i,l) \in T_W} p_d^i(\mathbf{x})\, h_i(\mathbf{z}_l^i | \mathbf{x}) \right) \left(\prod_{j:(j,*) \notin T_W} \phi_{g_j(\mathbf{x})} \right) d\mathbf{x}}{\displaystyle\prod_{(i,l) \in T_W} c_i(\mathbf{z}_l^i)}. \tag{C.27}$$

With these definitions we can prove, via mathematical induction, the following lemma.

Lemma 1. *Under the conditions of Assumption 2, the functional derivative of* $F[g_{1:s}, u]$ *with respect to the multisensor observation set* $Z_{k+1}^{1:s}$ *is given by*

$$\frac{\delta F}{\delta Z_{k+1}^{1:s}}[g_{1:s}, u] = \Gamma \sum_{P \in \mathcal{P}} \Psi_P[g_{1:s}, u] \prod_{W \in P} \varphi_W[g_{1:s}, u] \tag{C.28}$$

where Γ, $\Psi_P[g_{1:s}, u]$ *and* $\varphi_W[g_{1:s}, u]$ *are as defined in* (C.25), (C.26) *and* (C.27).

Lemma 1 is proved in Appendix C.4.

C.4 Proof of Lemma 1

The derivation is based on the approach used by Mahler [19] to derive multisensor PHD filter equations for the two sensor case and its extension by Delande et al. [122] for the general case of s sensors.

Proof. **Mathematical induction**

We prove using mathematical induction on $1 \leq \ell \leq s$ the following result,

$$\frac{\delta F}{\delta Z_{k+1}^{1:\ell}}[g_{1:s}, u] = \Gamma^{(\ell)} \sum_{P \in \mathcal{P}^{(\ell)}} \Psi_P^{(\ell)}[g_{1:s}, u] \prod_{W \in P} \varphi_W[g_{1:s}, u] \tag{C.29}$$

where,

$$\Gamma^{(\ell)} \stackrel{\text{def}}{=} \prod_{j=1}^{\ell} \prod_{\mathbf{z} \in Z_{k+1}^j} c_j(\mathbf{z}), \tag{C.30}$$

$$\Psi_P^{(\ell)}[g_{1:s}, u] \stackrel{\text{def}}{=} \left(\prod_{j=1}^{\ell} C_j^{(m_j - |P|_j)}(\langle c_j, g_j \rangle) \right) \left(\prod_{j=\ell+1}^{s} C_j(\langle c_j, g_j \rangle) \right) M^{(|P|-1)}(\langle r, u \prod_{j=1}^{s} \phi_{g_j} \rangle), \tag{C.31}$$

and for $W \in \mathcal{W}$,

$$\varphi_W[g_{1:s}, u] \stackrel{\text{def}}{=} \frac{\displaystyle\int_{\mathcal{X}} r(\mathbf{x}) u(\mathbf{x}) \left(\prod_{(i,l) \in T_W} p_d^i(\mathbf{x})\, h_i(\mathbf{z}_l^i | \mathbf{x}) \right) \left(\prod_{j:(j,*) \notin T_W} \phi_{g_j(\mathbf{x})} \right) d\mathbf{x}}{\displaystyle\prod_{(i,l) \in T_W} c_i(\mathbf{z}_l^i)}. \tag{C.32}$$

Mathematical induction: case $\ell = 1$

We first establish the induction result for the base case, i.e. $\ell = 1$. Ignoring the time index let the observation set gathered by sensor 1 at time $k+1$ be $Z_{k+1}^1 = \{\mathbf{z}_1^1, \mathbf{z}_2^1, \ldots, \mathbf{z}_{m_1}^1\}$. We have, for the case of s sensors

$$F[g_{1:s}, u] = \left(\prod_{j=1}^{s} C_j(\langle c_j, g_j \rangle) \right) M(\langle r, u \prod_{j=1}^{s} \phi_{g_j} \rangle). \tag{C.33}$$

Differentiating the above expression with respect to the set Z^1_{k+1} we get

$$\frac{\delta F}{\delta Z^1_{k+1}}[g_{1:s}, u] = \frac{\delta}{\delta Z^1_{k+1}} \left\{ \left(\prod_{j=1}^{s} C_j(\langle c_j, g_j \rangle) \right) M(\langle r, u \prod_{j=1}^{s} \phi_{g_j} \rangle) \right\} \tag{C.34}$$

$$= \left(\prod_{j=2}^{s} C_j(\langle c_j, g_j \rangle) \right) \frac{\delta}{\delta Z^1_{k+1}} \left\{ C_1(\langle c_1, g_1 \rangle)\, M(\langle r, u \prod_{j=1}^{s} \phi_{g_j} \rangle) \right\} \tag{C.35}$$

since the differential $\frac{\delta}{\delta Z^1_{k+1}}$ only differentiates the variable g_1. If $I \subseteq [\![1, m_1]\!]$ we can express $Y \subseteq Z^1_{k+1}$ as $Y = \{\mathbf{z}^1_i : i \in I\}$ for some I. We also have $Z^1_{k+1} - Y = \{\mathbf{z}^1_i : i \notin I\}$. Using the product rule for functional derivatives from (C.9) we have

$$\frac{\delta}{\delta Z^1_{k+1}} \left\{ C_1(\langle c_1, g_1 \rangle)\, M(\langle r, u \prod_{j=1}^{s} \phi_{g_j} \rangle) \right\}$$

$$= \sum_{n=0}^{m_1} \sum_{\substack{I \subseteq [\![1, m_1]\!] \\ |I| = n}} \frac{\delta}{\delta \{\mathbf{z}^1_i\}_{i \in I}} M(\langle r, u \prod_{j=1}^{s} \phi_{g_j} \rangle) \frac{\delta}{\delta \{\mathbf{z}^1_i\}_{i \notin I}} C_1(\langle c_1, g_1 \rangle). \tag{C.36}$$

Now we consider the derivatives of each of the individual terms in the above expression. By application of the chain rule for functional derivatives from (C.10)

$$\frac{\delta}{\delta \{\mathbf{z}^1_i\}_{i \in I}} M(\langle r, u \prod_{j=1}^{s} \phi_{g_j} \rangle) = M^{(n)}(\langle r, u \prod_{j=1}^{s} \phi_{g_j} \rangle) \prod_{i \in I} \langle r, u\, p^1_d\, h_1(\mathbf{z}^1_i) \prod_{j=2}^{s} \phi_{g_j} \rangle. \tag{C.37}$$

Similarly applying the chain rule to the second derivative

$$\frac{\delta}{\delta \{\mathbf{z}^1_i\}_{i \notin I}} C_1(\langle c_1, g_1 \rangle) = C_1^{(m_1 - n)}(\langle c_1, g_1 \rangle) \prod_{i \notin I} c_1(\mathbf{z}^1_i) \tag{C.38}$$

$$= C_1^{(m_1 - n)}(\langle c_1, g_1 \rangle) \frac{\Gamma^{(1)}}{\prod_{i \in I} c_1(\mathbf{z}^1_i)}. \tag{C.39}$$

Using these derivatives (C.36) can be expressed as

$$\frac{\delta}{\delta Z_{k+1}^1}\left\{C_1(\langle c_1, g_1\rangle)\, M(\langle r, u\prod_{j=1}^{s}\phi_{g_j}\rangle)\right\}$$
$$= \Gamma^{(1)}\sum_{n=0}^{m_1}\sum_{\substack{I\subseteq[\![1,m_1]\!]\\|I|=n}}\left\{M^{(n)}(\langle r, u\prod_{j=1}^{s}\phi_{g_j}\rangle)\, C_1^{(m_1-n)}(\langle c_1, g_1\rangle)\prod_{i\in I}\frac{\langle r, u\, p_d^1\, h_1(\mathbf{z}_i^1)\,\prod_{j=2}^{s}\phi_{g_j}\rangle}{c_1(\mathbf{z}_i^1)}\right\}. \tag{C.40}$$

In the double summation above, each set I maps to a partition P of the form $P = \bigcup_{i\in I}\{\mathbf{z}_i^1\}$ in $\mathcal{P}^{(1)}$ and vice versa. Hence using the result from equation (C.2) of Appendix C.1 we have

$$\frac{\delta F}{\delta Z_{k+1}^1}[g_{1:s}, u]$$
$$= \Gamma^{(1)}\sum_{P\in\mathcal{P}^{(1)}}\left\{C_1^{(m_1-|P|)}(\langle c_1, g_1\rangle)\left(\prod_{j=2}^{s}C_j(\langle c_j, g_j\rangle)\right)M^{(|P|-1)}(\langle r, u\prod_{j=1}^{s}\phi_{g_j}\rangle)\prod_{W\in P}\varphi_W[g_{1:s}, u]\right\} \tag{C.41}$$
$$= \Gamma^{(1)}\sum_{P\in\mathcal{P}^{(1)}}\Psi_P^{(1)}[g_{1:s}, u]\prod_{W\in P}\varphi_W[g_{1:s}, u]. \tag{C.42}$$

Hence the result is established for the case $\ell = 1$.

Mathematical induction: case $\ell = b \geq 1$

Now assuming that the result is true for some $\ell = b \geq 1$, we establish that the result holds for $\ell = b + 1 \leq s$. Let $Z_{k+1}^{b+1} = \{\mathbf{z}_1^{b+1}, \mathbf{z}_2^{b+1}, \ldots, \mathbf{z}_{m_{b+1}}^{b+1}\}$. We can write

$$\frac{\delta F}{\delta Z_{k+1}^{1:b+1}}[g_{1:s}, u] = \frac{\delta}{\delta Z_{k+1}^{b+1}}\left\{\frac{\delta F}{\delta Z_{k+1}^{1:b}}[g_{1:s}, u]\right\}. \tag{C.43}$$

Substituting the result for the case $\ell = b$ we get

$$\frac{\delta F}{\delta Z_{k+1}^{1:b+1}}[g_{1:s}, u] = \frac{\delta}{\delta Z_{k+1}^{b+1}}\left\{\Gamma^{(b)} \sum_{P \in \mathcal{P}^{(b)}} \Psi_P^{(b)}[g_{1:s}, u] \prod_{W \in P} \varphi_W[g_{1:s}, u]\right\} \tag{C.44}$$

$$= \Gamma^{(b)} \left(\prod_{j=b+2}^{s} C_j(\langle c_j, g_j \rangle)\right) \sum_{P \in \mathcal{P}^{(b)}} \left(\prod_{j=1}^{b} C_j^{(m_j - |P|_j)}(\langle c_j, g_j \rangle)\right)$$

$$\times \frac{\delta}{\delta Z_{k+1}^{b+1}}\left\{C_{b+1}(\langle c_{b+1}, g_{b+1} \rangle) M^{(|P|-1)}(\langle r, u \prod_{j=1}^{s} \phi_{g_j} \rangle) \prod_{W \in P} \varphi_W[g_{1:s}, u]\right\}. \tag{C.45}$$

Let $I_1 \subseteq [\![1, m_{b+1}]\!]$ and $I_2 \subseteq [\![1, m_{b+1}]\!]$ such that $I_1 \cap I_2 = \varnothing$. Then we can express $Y_1 \subseteq Z_{k+1}^{b+1}$ and $Y_2 \subseteq Z_{k+1}^{b+1}$ satisfying $Y_1 \cap Y_2 = \varnothing$ as $Y_1 = \{\mathbf{z}_i^{b+1} : i \in I_1\}$ and $Y_2 = \{\mathbf{z}_i^{b+1} : i \in I_2\}$ respectively. Applying the product rule from (C.8) to the expression above we have

$$\frac{\delta}{\delta Z_{k+1}^{b+1}}\left\{C_{b+1}(\langle c_{b+1}, g_{b+1} \rangle) M^{(|P|-1)}(\langle r, u \prod_{j=1}^{s} \phi_{g_j} \rangle) \prod_{W \in P} \varphi_W[g_{1:s}, u]\right\}$$

$$= \sum_{n_1=0}^{m_{b+1}} \sum_{n_2=0}^{\min(m_{b+1}, |P|-1)} \sum_{\substack{I_1 \subseteq [\![1, m_{b+1}]\!] \\ |I_1| = n_1}} \sum_{\substack{I_2 \subseteq [\![1, m_{b+1}]\!] \\ |I_2| = n_2;\ I_1 \cap I_2 = \varnothing}} \left\{\frac{\delta}{\delta\{\mathbf{z}_{i_1}^{b+1}\}_{i_1 \in I_1}} M^{(|P|-1)}(\langle r, u \prod_{n=1}^{s} \phi_{g_n} \rangle) \times \right.$$

$$\left. \frac{\delta}{\delta\{\mathbf{z}_{i_2}^{b+1}\}_{i_2 \in I_2}} \left(\prod_{W \in P} \varphi_W[g_{1:s}, u]\right) \frac{\delta}{\delta\{\mathbf{z}_i^{b+1}\}_{i \notin I_1 \cup I_2}} C_{b+1}(\langle c_{b+1}, g_{b+1} \rangle)\right\}. \tag{C.46}$$

The second summation above is restricted to the limit $\min(m_{b+1}, |P|-1)$ because the derivatives of $\prod_{W \in P} \varphi_W[g_{1:s}, u]$ for $n_2 > |P| - 1$ are zero. Now considering each of the individual derivatives above we have

$$\frac{\delta}{\delta\{\mathbf{z}_{i_1}^{b+1}\}_{i_1 \in I_1}} M^{(|P|-1)}(\langle r, u \prod_{n=1}^{s} \phi_{g_n} \rangle) = M^{(|P|+n_1-1)}(\langle r, u \prod_{n=1}^{s} \phi_{g_n} \rangle) \prod_{i_1 \in I_1} \varphi_{\{\mathbf{z}_{i_1}^{b+1}\}}[g_{1:s}, u]\, c_{b+1}(\mathbf{z}_{i_1}^{b+1}). \tag{C.47}$$

Denote $P = \{W_1, W_2, \ldots, W_{|P|-1}, V\}$ for notational convenience. Then we have

$$\frac{\delta}{\delta\{\mathbf{z}_{i_2}^{b+1}\}_{i_2\in I_2}}\left(\prod_{W\in P}\varphi_W[g_{1:s},u]\right) = \sum_{\substack{J\subseteq[\![1,|P|-1]\!]\\|J|=|I_2|}}\sum_{B\in\mathcal{B}(I_2,J)}\left\{\left(\prod_{j\notin J}\varphi_{W_j}[g_{1:s},u]\right)\left(\prod_{i_2\in I_2}\varphi_{W_{i_2}^B}[g_{1:s},u]\, c_{b+1}(\mathbf{z}_{i_2}^{b+1})\right)\right\} \tag{C.48}$$

where $\mathcal{B}(I_2, J)$ is the collection of all possible matchings from set I_2 to set J and we define the measurement subset $W_{i_2}^B \stackrel{\text{def}}{=} W_{B(i_2)} \cup \mathbf{z}_{i_2}^{b+1}$. Also

$$\frac{\delta}{\delta\{\mathbf{z}_i^{b+1}\}_{i\notin I_1\cup I_2}}\left(C_{b+1}(\langle c_{b+1}, g_{b+1}\rangle)\right) = C_{b+1}^{(m_{b+1}-n_1-n_2)}(\langle c_{b+1}, g_{b+1}\rangle)\prod_{i\notin I_1\cup I_2} c_{b+1}(\mathbf{z}_i^{b+1}). \tag{C.49}$$

Combining the three derivatives into expression (C.45) we get

$$\begin{aligned}\frac{\delta F}{\delta Z_{k+1}^{1:b+1}}[g_{1:s},u] = \Gamma^{(b)}&\left(\prod_{j=b+2}^{s} C_j(\langle c_j, g_j\rangle)\right)\left(\prod_{\mathbf{z}^{b+1}\in Z_{k+1}^{b+1}} c_{b+1}(\mathbf{z}^{b+1})\right)\times\\ &\sum_{P\in\mathcal{P}^{(b)}}\sum_{n_1=0}^{m_{b+1}}\sum_{n_2=0}^{\min(m_{b+1},|P|-1)}\sum_{\substack{I_1\subseteq[\![1,m_{b+1}]\!]\\|I_1|=n_1}}\sum_{\substack{I_2\subseteq[\![1,m_{b+1}]\!]\\|I_2|=n_2;\, I_1\cap I_2=\varnothing}}\sum_{\substack{J\subseteq[\![1,|P|-1]\!]\\|J|=|I_2|}}\sum_{B\in\mathcal{B}(I_2,J)}\\ &\left\{C_{b+1}^{(m_{b+1}-n_1-n_2)}(\langle c_{b+1}, g_{b+1}\rangle)\left(\prod_{j\notin J}\varphi_{W_j}[g_{1:s},u]\right)M^{(|P|+n_1-1)}(\langle r, u\prod_{n=1}^{s}\phi_{g_n}\rangle)\times\right.\\ &\left.\left(\prod_{j=1}^{b} C_j^{(m_j-|P|_j)}(\langle c_j, g_j\rangle)\right)\left(\prod_{i_1\in I_1}\varphi_{\{\mathbf{z}_{i_1}^{b+1}\}}[g_{1:s},u]\right)\left(\prod_{i_2\in I_2}\varphi_{W_{i_2}^B}[g_{1:s},u]\right)\right\}.\end{aligned} \tag{C.50}$$

Using result of Appendix C.1 we can simplify the multiple summation term and write

$$\begin{aligned}\frac{\delta F}{\delta Z_{k+1}^{1:b+1}}[g_{1:s},u] &= \Gamma^{(b+1)}\sum_{P\in\mathcal{P}^{(b+1)}}\left\{\left(\prod_{j=1}^{b+1} C_j^{(m_j-|P|_j)}(\langle c_j, g_j\rangle)\right)\left(\prod_{j=b+2}^{s} C_j(\langle c_j, g_j\rangle)\right)\times\right.\\ &\qquad\left.M^{(|P|+n_1-1)}(\langle r, u\prod_{n=1}^{s}\phi_{g_n}\rangle)\left(\prod_{W\in P}\varphi_W[g_{1:s},u]\right)\right\} &\text{(C.51)}\\ &= \Gamma^{(b+1)}\sum_{P\in\mathcal{P}^{(b+1)}}\Psi_P^{(b+1)}[g_{1:s},u]\prod_{W\in P}\varphi_W[g_{1:s},u]. &\text{(C.52)}\end{aligned}$$

Hence we have established the result stated in (C.29) using the method of mathematical induction. We obtain the result of Lemma 1 by substituting $\ell = s$ in this result.

□

C.5 Proof of Theorem 1

Proof. For brevity denote $\Psi_P[0,0,\ldots,0,u] = \Psi_P[u]$ and $\varphi_W[0,0,\ldots,0,u] = \varphi_W[u]$. Substituting $g_j \equiv 0$ for $j = 1,2,\ldots,s$ in the result of Lemma 1 we get

$$\frac{\delta F}{\delta Z_{k+1}^{1:s}}[0,0,\ldots,0,u] = \Gamma \sum_{P\in\mathcal{P}} \Psi_P[u] \prod_{W\in P} \varphi_W[u]. \tag{C.53}$$

PHD update

Differentiating equation (C.53) with respect to set $\{\mathbf{x}\}$ we have

$$\frac{\delta F}{\delta \mathbf{x}\delta Z_{k+1}^{1:s}}[0,0,\ldots,0,u] = \frac{\delta}{\delta \mathbf{x}}\left\{\frac{\delta F}{\delta Z_{k+1}^{1:s}}[0,0,\ldots,0,u]\right\} \tag{C.54}$$

$$= \frac{\delta}{\delta \mathbf{x}}\left\{\Gamma \sum_{P\in\mathcal{P}} \Psi_P[u] \prod_{W\in P} \varphi_W[u]\right\} \tag{C.55}$$

$$= \Gamma \sum_{P\in\mathcal{P}} \left(\prod_{j=1}^{s} C_j^{(m_j-|P|_j)}(0)\right) \frac{\delta}{\delta \mathbf{x}}\left\{M^{(|P|-1)}(\langle r, u\prod_{j=1}^{s} q_d^j\rangle) \prod_{W\in P} \varphi_W[u]\right\}. \tag{C.56}$$

Applying the product rule for functional derivatives from (C.9)

$$\frac{\delta}{\delta \mathbf{x}}\left\{M^{(|P|-1)}(\langle r, u\prod_{j=1}^{s} q_d^j\rangle) \prod_{W\in P} \varphi_W[u]\right\}$$
$$= \frac{\delta}{\delta \mathbf{x}}\left\{M^{(|P|-1)}(\langle r, u\prod_{j=1}^{s} q_d^j\rangle)\right\} \prod_{W\in P} \varphi_W[u] + M^{(|P|-1)}(\langle r, u\prod_{j=1}^{s} q_d^j\rangle)\frac{\delta}{\delta \mathbf{x}}\left\{\prod_{W\in P} \varphi_W[u]\right\}. \tag{C.57}$$

Evaluating the individual derivatives above and substituting the constant function $u(\mathbf{x}) \equiv 1$, we get

$$\frac{\delta}{\delta \mathbf{x}} \left\{ M^{(|P|-1)}(\langle r, u \prod_{j=1}^{s} q_d^j \rangle) \right\}_{u \equiv 1} = M^{(|P|)}(\gamma)\, r(\mathbf{x}) \prod_{j=1}^{s} q_d^j(\mathbf{x}) \tag{C.58}$$

$$\frac{\delta}{\delta \mathbf{x}} \left\{ \prod_{W \in P} \varphi_W[u] \right\}_{u \equiv 1} = \left(\prod_{W \in P} d_W \right) \left(\sum_{W \in P} r(\mathbf{x}) \rho_W(\mathbf{x}) \right) \tag{C.59}$$

where γ, d_W and $\rho_W(\mathbf{x})$ are defined in (6.41), (6.43) and (6.44) respectively.

Hence we have

$$\begin{aligned} &\frac{\delta F}{\delta \mathbf{x} \delta Z_{k+1}^{1:s}}[0, 0, \ldots, 0, 1] = \Gamma \sum_{P \in \mathcal{P}} \left(\prod_{j=1}^{s} C_j^{(m_j - |P|_j)}(0) \right) \times \\ &\left\{ M^{(|P|)}(\gamma)\, r(\mathbf{x}) \prod_{j=1}^{s} q_d^j(\mathbf{x}) \left(\prod_{W \in P} d_W \right) + M^{(|P|-1)}(\gamma) \left(\prod_{W \in P} d_W \right) \left(\sum_{W \in P} r(\mathbf{x}) \rho_W(\mathbf{x}) \right) \right\} \end{aligned} \tag{C.60}$$

$$= \Gamma \sum_{P \in \mathcal{P}} \left(\kappa_P M^{(|P|)} \prod_{W \in P} d_W \right) \left(r(\mathbf{x}) \prod_{j=1}^{s} q_d^j(\mathbf{x}) \right) + \Gamma \sum_{P \in \mathcal{P}} \left(\kappa_P M^{(|P|-1)} \prod_{W \in P} d_W \right) \left(\sum_{W \in P} r(\mathbf{x})\, \rho_W(\mathbf{x}) \right) \tag{C.61}$$

where κ_P is defined in (6.42). Substituting $u(\mathbf{x}) \equiv 1$ in equation (C.53) we have

$$\frac{\delta F}{\delta Z_{k+1}^{1:s}}[0, 0, \ldots, 0, 1] = \Gamma \sum_{P \in \mathcal{P}} \kappa_P M^{(|P|-1)} \prod_{W \in P} d_W. \tag{C.62}$$

Dividing (C.61) by (C.62) and using the definition of PHD from (C.24), we get

$$D_{k+1|k+1}(\mathbf{x}) = \frac{\dfrac{\delta F}{\delta \mathbf{x} \delta Z_{k+1}^{1:s}}[0, 0, \ldots, 0, 1]}{\dfrac{\delta F}{\delta Z_{k+1}^{1:s}}[0, 0, \ldots, 0, 1]} \tag{C.63}$$

$$= r(\mathbf{x}) \left\{ \alpha_0 \prod_{j=1}^{s} q_d^j(\mathbf{x}) + \sum_{P \in \mathcal{P}} \alpha_P \left(\sum_{W \in P} \rho_W(\mathbf{x}) \right) \right\} \tag{C.64}$$

where α_0 and α_P are as given in (6.45) and (6.46).

Cardinality update

We now derive the update equation for the posterior cardinality distribution. Using the expression for the posterior probability generating functional in (C.23) and the results of (C.53) and (C.62) we have

$$G_{k+1|k+1}[u] = \frac{\sum_{P\in\mathcal{P}} \Psi_P[u] \prod_{W\in P} \varphi_W[u]}{\sum_{P\in\mathcal{P}} \kappa_P M^{(|P|-1)} \prod_{W\in P} d_W}. \tag{C.65}$$

The probability generating function $M_{k+1|k+1}(t)$ of the posterior cardinality distribution is obtained by substituting the constant function $u(\mathbf{x}) \equiv t$ in the expression for $G_{k+1|k+1}[u]$. Thus

$$M_{k+1|k+1}(t) = \frac{\sum_{P\in\mathcal{P}} \Psi_P[t] \prod_{W\in P} \varphi_W[t]}{\sum_{P\in\mathcal{P}} \kappa_P M^{(|P|-1)} \prod_{W\in P} d_W}. \tag{C.66}$$

For constant t we have

$$\prod_{W\in P} \varphi_W[t] = t^{|P|-1} \prod_{W\in P} d_W \tag{C.67}$$

$$\Psi_P[t] = \left(\prod_{j=1}^{s} C_j^{(m_j-|P|_j)}(0)\right) M^{(|P|-1)}(t\gamma). \tag{C.68}$$

Since $M_{k+1|k+1}(t)$ is the PGF corresponding to the cardinality distribution $\pi_{k+1|k+1}(n)$,

$$\pi_{k+1|k+1}(n) = \frac{1}{n!} M^{(n)}_{k+1|k+1}(0) \tag{C.69}$$

$$= \frac{1}{n!} \left\{ \frac{d^n \sum_{P\in\mathcal{P}} \Psi_P[t] t^{|P|-1} \prod_{W\in P} d_W}{dt^n \sum_{P\in\mathcal{P}} \kappa_P M^{(|P|-1)} \prod_{W\in P} d_W} \right\}_{t=0} \tag{C.70}$$

$$= \frac{\sum_{P\in\mathcal{P}} \left(\prod_{j=1}^{s} C_j^{(m_j-|P|_j)}(0)\right) \prod_{W\in P} d_W \left\{ \frac{d^n}{dt^n} t^{|P|-1} M^{(|P|-1)}(t\gamma) \right\}_{t=0}}{n! \sum_{P\in\mathcal{P}} \kappa_P M^{(|P|-1)} \prod_{W\in P} d_W}. \tag{C.71}$$

Evaluating the derivative we get

$$\left\{\frac{d^n}{dt^n}t^{|P|-1}\,M^{(|P|-1)}(t\gamma)\right\}_{t=0} = \begin{cases} 0 & \text{if } n < |P|-1 \\ \dfrac{n!}{(n-|P|+1)!}M^{(n)}(0)\,\gamma^{n-|P|+1} & \text{if } n \geq |P|-1. \end{cases} \tag{C.72}$$

We also have $M^{(n)}(0) = n!\,\pi_{k+1|k}(n)$, hence

$$\pi_{k+1|k+1}(n) = \pi_{k+1|k}(n)\,\frac{\sum\limits_{\substack{P\epsilon\mathcal{P} \\ |P|\leq n+1}} \dfrac{n!}{(n-|P|+1)!}\left(\prod\limits_{j=1}^{s} C_j^{(m_j-|P|_j)}(0)\right)\gamma^{n-|P|+1}\prod\limits_{W\epsilon P} d_W}{\sum\limits_{P\epsilon\mathcal{P}} \kappa_P M^{(|P|-1)}\prod\limits_{W\epsilon P} d_W}. \tag{C.73}$$

We thus have

$$\frac{\pi_{k+1|k+1}(n)}{\pi_{k+1|k}(n)} = \frac{\sum\limits_{\substack{P\epsilon\mathcal{P} \\ |P|\leq n+1}} \left(\kappa_P \dfrac{n!}{(n-|P|+1)!}\gamma^{n-|P|+1}\prod\limits_{W\epsilon P} d_W\right)}{\sum\limits_{P\epsilon\mathcal{P}} \kappa_P M^{(|P|-1)}\prod\limits_{W\epsilon P} d_W} \tag{C.74}$$

where κ_P is as defined in (6.42).

□

References

[1] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *Radar and Signal Proc., IEE Proc. F*, vol. 140, pp. 107–113, Apr. 1993.

[2] M. Bugallo, T. Lu, and P. Djuric, "Target tracking by multiple particle filtering," in *Proc. IEEE Aerospace Conf.*, (Big Sky, MT), Mar. 2007.

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

[4] F. Thouin, S. Nannuru, and M. Coates, "Multi-target tracking for measurement models with additive contributions," in *Proc. Int. Conf. Information Fusion*, (Chicago, IL, U.S.A.), Jul. 2011.

[5] J. Wilson and N. Patwari, "Radio tomographic imaging with wireless networks," *IEEE Trans. Mobile Comp.*, vol. 9, pp. 621–632, May 2010.

[6] J. Wilson and N. Patwari, "See through walls: motion tracking using variance-based radio tomography networks," *IEEE Trans. Mobile Computing*, vol. 10, pp. 612–621, May 2011.

[7] Y. Li, X. Chen, M. Coates, and B. Yang, "Sequential Monte Carlo radio-frequency tomographic tracking," in *Proc. Int. Conf. Acoustics, Speech and Signal Proc.*, (Prague, Czech Republic), May 2011.

[8] X. Chen, A. Edelstein, Y. Li, M. Coates, M. Rabbat, and A. Men, "Sequential Monte Carlo for simultaneous passive device-free tracking and sensor localization using received signal strength measurements," in *Proc. Int. Conf. Inf. Proc. Sens. Networks*, (Chicago, IL, U.S.A.), Apr. 2011.

[9] R. Mahler, ""Statistics 101" for multisensor, multitarget data fusion," *IEEE Aerospace and Electronic Systems Magazine*, vol. 19, pp. 53–64, Jan. 2004.

[10] O. Hlinka, O. Sluciak, F. Hlawatsch, P. M. Djuric, and M. Rupp, "Likelihood consensus and its application to distributed particle filtering," *IEEE Trans. Sig. Proc.*, vol. 60, pp. 4334–4349, Aug. 2012.

[11] B. Balakumar, A. Sinha, T. Kirubarajan, and J. Reilly, "PHD filtering for tracking an unknown number of sources using an array of sensors," in *Proc. Workshop Stat. Sig. Proc.*, (Bordeaux, France), Jul. 2005.

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

[13] D. Angelosante, E. Biglieri, and M. Lops, "Multipath channel tracking in OFDM systems," in *IEEE Int. Symposium Personal, Indoor and Mobile Radio Communications*, (Athens, Greece), Sep. 2007.

[14] R. Mahler, "CPHD filters for superpositional sensors," in *Proc. SPIE Int. Conf. Sig. Data Proc. Small Targets*, (San Diego, CA, U.S.A.), Aug. 2009.

[15] D. Hauschildt, "Gaussian mixture implementation of the cardinalized probability hypothesis density filter for superpositional sensors," in *Proc. Int. Conf. Indoor Positioning and Indoor Navigation*, (Guimaraes, Portugal), Sep. 2011.

[16] S. S. Blackman, *Multiple-target tracking with radar applications*. Artech House, Boston, 1986.

[17] D. Clark, I. Ruiz, Y. Petillot, and J. Bell, "Particle PHD filter multiple target tracking in sonar image," *IEEE Trans. Aerospace and Elec. Sys.*, vol. 43, pp. 409–416, Jan. 2007.

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

[19] R. Mahler, "The multisensor PHD filter: I. General solution via multitarget calculus," in *Proc. SPIE Int. Conf. Sig. Proc., Sensor Fusion, Target Recog.*, (Orlando, FL, U.S.A.), Apr. 2009.

[20] R. Mahler, "The multisensor PHD filter: II. Erroneous solution via Poisson magic," in *Proc. SPIE Int. Conf. Sig. Proc., Sensor Fusion, Target Recog.*, (Orlando, FL, U.S.A.), Apr. 2009.

[21] R. Mahler, "Approximate multisensor CPHD and PHD filters," in *Proc. Int. Conf. Inf. Fusion*, (Edinburgh, U.K.), Jul. 2010.

[22] S. Nannuru, Y. Li, Y. Zeng, M. Coates, and B. Yang, "Radio frequency tomography for passive indoor multi-target tracking," *IEEE Trans. Mobile Comp.*, vol. 12, pp. 2322–2333, Dec. 2013.

[23] S. Nannuru, Y. Li, M. Coates, and B. Yang, "Multi-target device-free tracking using radio frequency tomography," in *Proc. Int. Conf. Intell. Sensors, Sens. Net. and Inf. Processing*, (Adelaide, Australia), Dec. 2011.

[24] S. Nannuru, M. Coates, and R. Mahler, "Computationally-tractable approximate PHD and CPHD filters for superpositional sensors," *IEEE J. Sel. Topics in Sig. Proc.*, vol. 7, pp. 410–420, Jun. 2013.

[25] S. Nannuru and M. Coates, "Multi-Bernoulli filter for superpositional sensors," in *Proc. Int. Conf. Inf. Fusion*, (Istanbul, Turkey), Jul. 2013.

[26] S. Nannuru and M. Coates, "Particle filter implementation of the multi-Bernoulli filter for superpositional sensors," in *Proc. Int. Workshop Comp. Adv. Multi-Sensor Adap. Processing*, (Saint Martin), Dec. 2013.

[27] S. Nannuru and M. Coates, "Hybrid multi-Bernoulli CPHD filter for superpositional sensors," in *Proc. SPIE Int. Conf. Sig. Proc., Sensor Fusion, Target Recog.*, (Baltimore, MD, U.S.A.), May 2014.

[28] S. Nannuru and M. Coates, "Hybrid multi-Bernoulli and cphd filters for superpositional sensors," *accepted in IEEE Trans. Aerospace and Elec. Sys.*, 2015.

[29] S. Nannuru, M. Coates, M. Rabbat, and S. Blouin, "General solution and approximate implementation of the multisensor multitarget CPHD filter," in *Proc. Int. Conf. Acoustics, Speech and Signal Proc.*, (Brisbane,Australia), Apr. 2015.

[30] S. Nannuru, S. Blouin, M. Coates, and M. Rabbat, "Multisensor CPHD filter," *submited to IEEE Trans. Aerospace and Elec. Sys.*, 2015.

[31] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE J. Oceanic Engineering*, vol. 8, pp. 173–184, Jul. 1983.

[32] D. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. on Automatic Control*, vol. 24, pp. 843–854, Dec. 1979.

[33] I. R. Goodman, R. Mahler, and H. T. Nguyen, *Mathematics of data fusion*. Boston, U.S.A.: Springer, 1997.

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

[35] R. Mahler, *Advances in statistical multisource-multitarget information fusion*. Artech House, Boston, 2014.

[36] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo methods for multitarget filtering with random finite sets," *IEEE Trans. Aerospace and Electronic Systems*, vol. 41, pp. 1224–1245, Oct. 2005.

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

[38] R. Mahler and A. El-Fallah, "An approximate CPHD filter for superpositional sensors," in *Proc. SPIE Int. Conf. Sig. Proc., Sensor Fusion, Target Recog.*, (Baltimore, MD, U.S.A.), Apr. 2012.

[39] B. Vo, B. Vo, and A. Cantoni, "The cardinality balanced multi-target multi-Bernoulli filter and its implementations," *IEEE Trans. Signal Proc.*, vol. 57, pp. 409–423, Feb. 2009.

[40] L. D. Stone, T. L. Corwin, and C. A. Barlow, *Bayesian Multiple Target Tracking*. 1999.

[41] B.-N. Vo and W.-K. Ma, "The Gaussian mixture probability hypothesis density filter," *IEEE Trans. Sig. Proc.*, vol. 54, pp. 4091–4104, Nov. 2006.

[42] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Trans. Sig. Proc.*, vol. 56, pp. 3447–3457, Aug. 2008.

[43] T. Teixeira, G. Dublon, and A. Savvides, "A survey of human-sensing: Methods for detecting presence, count, location, track, and identity," *ACM Computing Surveys*, vol. 5, Sep. 2010.

[44] A. Lipton, H. Fujiyoshi, and R. Patil, "Moving target classification and tracking from real-time video," in *Proc. Int. Workshop Applications of Computer Vision*, (Princeton, NJ, U.S.A.), Oct. 1998.

[45] M. Enzweiler and D. Gavrila, "Monocular pedestrian detection: Survey and experiments," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, pp. 2179–2195, Dec. 2009.

[46] D. Schulz, D. Fox, and J. Hightower, "People tracking with anonymous and id-sensors using Rao-Blackwellised particle filters," in *Proc. Int. Joint Conf. Artificial Intelligence*, (Acapulco, Mexico), Aug. 2003.

[47] M. Bertozzi, A. Broggi, C. Caraffi, M. D. Rose, M. Felisa, and G. Vezzoni, "Pedestrian detection by means of far-infrared stereo vision," *Computer Vision and Image Understanding*, vol. 106, pp. 194 – 204, May 2007.

[48] X. Sheng and Y.-H. Hu, "Energy based acoustic source localization," in *Proc. Information Processing in Sensor Networks*, (Palo Alto, CA, U.S.A.), Apr. 2003.

[49] W. Xiao, J. K. Wu, L. Shue, Y. Li, and L. Xie, "A prototype ultrasonic sensor network for tracking of moving targets," in *IEEE Conf. Industrial Elec. and Appl.*, (Singapore), May 2006.

[50] L. Ni, Y. Liu, Y. Lau, and A. Patil, "LANDMARC: Indoor location sensing using active RFID," *Kluwer Academic Publishers: Wireless Networks*, vol. 10, pp. 701–710, Nov. 2004.

[51] S. Chang, R. Sharan, M. Wolf, N. Mitsumoto, and J. Burdick, "UWB radar-based human target tracking," in *IEEE Radar Conference*, (Pasadena, CA, U.S.A.), May 2009.

[52] M. Youssef and A. Agrawala, "The Horus location determination system," *Springer-Verlag Wireless Networks*, vol. 14, pp. 357–374, Jun. 2008.

[53] N. Patwari and J. Wilson, "RF sensor networks for device-free localization: measurements, models, and algorithms," *Proc. IEEE*, vol. 98, pp. 1961–1973, Nov. 2010.

[54] D. Zhang, J. Ma, Q. Chen, and L. Ni, "An RF-based system for tracking transceiver-free objects," in *Proc. IEEE Int. Conf. Perv. Comp. and Comm.*, (White Plains, NY, U.S.A.), Mar. 2007.

[55] D. Zhang, J. Ma, Q. Chen, and L. Ni, "Dynamic clustering for tracking multiple transceiver-free objects," in *Proc. IEEE Int. Conf. Perv. Comp. Comm.*, (Galveston, TX, U.S.A.), Mar. 2009.

[56] D. Zhang, Y. Liu, and L. Ni, "RASS: A real-time, accurate and scalable system for tracking transceiver-free objects," in *Proc. IEEE Int. Conf. Perv. Comp. Comm.*, (Seattle, WA, U.S.A.), Mar. 2011.

[57] M. Youssef, M. Mah, and A. Agrawala, "Challenges: device-free passive localization for wireless environments," in *Proc. Int. Conf. Mobile Computing and Networking*, (Montreal, QC, Canada), Sep. 2007.

[58] M. Moussa and M. Youssef, "Smart devices for smart environments: device-free passive detection in real environments," in *Proc. Int. Conf. Perv. Comp. Comm.*, (Galveston, TX, U.S.A.), Mar. 2009.

[59] M. Kanso and M. Rabbat, "Compressed RF tomography for wireless sensor networks: Centralized and decentralized approaches," in *Proc. IEEE Dist. Computing in Sensor Systems*, (Santa Barbara, CA, U.S.A.), Jun. 2010.

[60] J. Wilson and N. Patwari, "A fade level skew-Laplace signal strength model for device-free localization with wireless networks," *IEEE Trans. Mobile Computing*, vol. 11, pp. 947–958, Jun. 2012.

[61] R. E. Kalman, "A new approach to linear filtering and prediction problems," *ASME J. Fluids Engineering*, vol. 82, pp. 35–45, Mar. 1960.

[62] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Proc. Int. Conf. AeroSense*, (Orlando, FL, U.S.A.), Apr. 1997.

[63] S. Julier and J. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, pp. 401–422, Mar. 2004.

[64] M. K. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *J. American statistical association*, vol. 94, pp. 590–599, Jun. 1999.

[65] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Sig. Proc.*, vol. 50, pp. 174–188, Feb. 2002.

[66] A. Doucet, N. De Freitas, and N. Gordon, *An introduction to sequential Monte Carlo methods*. Springer, New York, 2001.

[67] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman filter: Particle filters for tracking applications*. Artech House, Boston, 2004.

[68] C. Hue, J.-P. Le Cadre, and P. Perez, "Sequential Monte Carlo methods for multiple target tracking and data fusion," *IEEE Trans. Signal Processing*, vol. 50, pp. 309–325, Feb. 2002.

[69] T. Zhao and R. Nevatia, "Tracking multiple humans in crowded environment," in *IEEE Proc. Conf. Computer Vision and Pattern Recognition*, (Washington, DC, U.S.A.), Jun. 2004.

[70] Z. Khan, T. Balch, and F. Dellaert, "MCMC-based particle filtering for tracking a variable number of interacting targets," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 1805–1918, Nov. 2005.

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

[72] F. Daum and J. Huang, "Curse of dimensionality and particle filters," in *IEEE Proc. Aerospace Conference*, (Big Sky, MT, U.S.A.), Mar. 2003.

[73] P. Djuric and M. Bugallo, "Improved target tracking with particle filtering," in *Proc. Int. Aerospace Conf.*, (Big Sky, MT, U.S.A.), Mar. 2009.

[74] M. Bugallo and P. Djuric, "Target tracking by symbiotic particle filtering," in *Proc. Int. Aerospace Conf.*, (Big Sky, MT, U.S.A.), Mar. 2010.

[75] P. Djuric and M. Bugallo, "Adaptive systems of particle filters," in *Proc. Asilomar Conf. Sig., Sys. and Computers*, (Pacific Grove, CA, U.S.A.), Nov. 2010.

[76] P. Closas and M. Bugallo, "Improving accuracy by iterated multiple particle filtering," *Signal Processing Letters*, vol. 19, pp. 531–534, Aug. 2012.

[77] P. Djuric and M. Bugallo, "Particle filtering for high-dimensional systems," in *Proc. Int. Workshop Comp. Advances in Multi-Sensor Adaptive Proc.*, (Saint Martin), Dec. 2013.

[78] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Artech House, Boston, 1999.

[79] N. Bergman and A. Doucet, "Markov chain Monte Carlo data association for target tracking," in *Proc. Int. Conf. Acoustics, Speech and Signal Proc.*, (Istanbul, Turkey), Jun. 2000.

[80] D. Schulz, W. Burgard, D. Fox, and A. Cremers, "Tracking multiple moving targets with a mobile robot using particle filters and statistical data association," in *Proc. Int. Conf. Robotics and Automation*, (Seoul, Korea), May 2001.

[81] J. Vermaak, S. Godsill, and P. Perez, "Monte Carlo filtering for multi target tracking and data association," *IEEE Trans. Aerospace and Elec. Sys.*, vol. 41, pp. 309–332, Jan. 2005.

[82] S. Oh, S. Russell, and S. Sastry, "Markov chain Monte Carlo data association for multi-target tracking," *IEEE Trans. Automatic Control*, vol. 54, pp. 481–497, Mar. 2009.

[83] M. Vihola, "Random sets for multitarget tracking and data fusion," *Licentiate thesis*, 2004.

[84] W.-K. Ma, B.-N. Vo, S. Singh, and A. Baddeley, "Tracking an unknown time-varying number of speakers using TDOA measurements: A random finite set approach," *IEEE Trans. Sig. Proc.*, vol. 54, pp. 3291–3304, Sep. 2006.

[85] B.-T. Vo, B.-N. Vo, and A. Cantoni, "Bayesian filtering with random finite set observations," *IEEE Trans. Signal Proc.*, vol. 56, pp. 1313–1326, Apr. 2008.

[86] R. Mahler, "Multitarget Bayes filtering via first-order multitarget moments," *IEEE Trans. Aerospace and Electronic Systems*, vol. 39, pp. 1152–1178, Oct. 2003.

[87] R. Mahler, "PHD filters of higher order in target number," *IEEE Trans. Aerospace and Elec. Sys.*, vol. 43, pp. 1523–1543, Oct. 2007.

[88] S. Reuter, B. Vo, B. Vo, and K. Dietmayer, "The labeled multi-Bernoulli filter," *IEEE Trans. Signal Proc.*, vol. 62, pp. 3246–3260, Jun. 2014.

[89] B. Vo and B. Vo, "Labeled random finite sets and multi-object conjugate priors," *IEEE Trans. Signal Proc.*, vol. 61, pp. 3460–3475, Jul. 2013.

[90] B.-T. Vo, B.-N. Vo, and A. Cantoni, "Analytic implementations of the cardinalized probability hypothesis density filter," *IEEE Trans. Sig. Proc.*, vol. 55, pp. 3553–3567, Jul. 2007.

[91] B. T. Vo, *Random finite sets in multi-object filtering*. PhD thesis, The University of Western Australia, Perth, Australia, 2008.

[92] K. Panta, D. Clark, and B.-N. Vo, "Data association and track management for the Gaussian mixture probability hypothesis density filter," *IEEE Trans. Aerospace and Elec. Sys.*, vol. 45, pp. 1003–1016, Jul. 2009.

[93] S. Pasha, B.-N. Vo, H. D. Tuan, and W.-K. Ma, "A Gaussian mixture PHD filter for jump Markov system models," *IEEE Trans. Aerospace and Elec. Sys.*, vol. 45, pp. 919–936, Jul. 2009.

[94] K. Panta, *Multi-target tracking using 1st moment of random finite sets*. PhD thesis, University of Melbourne, Melbourne, Australia, 2008.

[95] T. Zajic and R. P. Mahler, "Particle-systems implementation of the PHD multitarget-tracking filter," in *Proc. Int. Conf. AeroSense*, (Orlando, FL, U.S.A.), Apr. 2003.

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

[97] L. Lin, Y. Bar-Shalom, and T. Kirubarajan, "Track labeling and PHD filter for multitarget tracking," *IEEE Trans. Aerospace and Elec. Sys.*, vol. 42, pp. 778–795, Jul. 2006.

[98] B. Ristic, *Particle filters for random set models*. Springer, New York, 2013.

[99] D. Clark and J. Bell, "Convergence results for the particle PHD filter," *IEEE Trans. Signal Processing*, vol. 54, pp. 2652–2661, Jul. 2006.

[100] A. M. Johansen, S. S. Singh, A. Doucet, and B.-N. Vo, "Convergence of the SMC implementation of the PHD filter," *J. Methodology and Computing in Applied Probability*, vol. 8, pp. 265–291, Jun. 2006.

[101] D. Clark and B.-N. Vo, "Convergence analysis of the Gaussian mixture PHD filter," *IEEE Trans. Signal Processing*, vol. 55, pp. 1204–1212, Apr. 2007.

[102] D. E. Clark, *Multiple target tracking with the probability hypothesis density filter*. PhD thesis, Heriot-Watt University, Edinburgh, U.K., 2006.

[103] F. Lian, C. Li, C. Han, and H. Chen, "Convergence analysis for the SMC-MeMBer and SMC-CBMeMBer filters," *J. of App. Mathematics*, vol. 2012, 2012.

[104] B. Ristic, D. Clark, and B.-N. Vo, "Improved SMC implementation of the PHD filter," in *Proc. Int. Conf. Inf. Fusion*, (Edinburgh, U.K.), Jul. 2010.

[105] N. Whiteley, S. Singh, and S. Godsill, "Auxiliary particle implementation of probability hypothesis density filter," *IEEE Trans. Aerospace and Electronic Systems*, vol. 46, pp. 1437–1454, Jul. 2010.

[106] E. Baser and M. Efe, "A novel auxiliary particle PHD filter," in *Proc. Int. Conf. Inf. Fusion*, (Singapore), Jul. 2012.

[107] B. Ristic, D. Clark, B.-N. Vo, and B.-T. Vo, "Adaptive target birth intensity for PHD and CPHD filters," *IEEE Trans. Aerospace and Elec. Sys.*, vol. 48, pp. 1656–1668, Apr. 2012.

[108] D. Clark, A.-T. Cemgil, P. Peeling, and S. Godsill, "Multi-object tracking of sinusoidal components in audio with the Gaussian mixture probability hypothesis density filter," in *Proc. Int. Workshop Applications of Signal Processing to Audio and Acoustics*, (New Paltz, NY, U.S.A.), Oct. 2007.

[109] D. Clark and S. Godsill, "Group target tracking with the Gaussian mixture probability hypothesis density filter," in *Proc. Int. Conf. Intell. Sensors, Sens. Net. and Inf. Processing*, (Melbourne, Australia), Dec. 2007.

[110] M. Ulmke, O. Erdinc, and P. Willett, "Gaussian mixture cardinalized PHD filter for ground moving target tracking," in *Proc. Int. Conf. Information Fusion*, (Québec city, QC, Canada), Jul. 2007.

[111] R. Mahler, "PHD filters for nonstandard targets, I: Extended targets," in *Proc. Int. Conf. Inf. Fusion*, (Seattle, WA, U.S.A.), Jul. 2009.

[112] K. Granstrom, C. Lundquist, and U. Orguner, "A Gaussian mixture PHD filter for extended target tracking," in *Proc. Int. Conf. Inf. Fusion*, (Edinburgh, U.K.), Jul. 2010.

[113] J. Mullane, B.-N. Vo, M. Adams, and B.-T. Vo, "A random-finite-set approach to Bayesian SLAM," *IEEE Trans. Robotics*, vol. 27, pp. 268–282, Apr. 2011.

[114] C. Lundquist, L. Hammarstrand, and F. Gustafsson, "Road intensity based mapping using radar measurements with a probability hypothesis density filter," *IEEE Trans. Signal Processing*, vol. 59, pp. 1397–1408, Apr. 2011.

[115] E. Maggio, E. Piccardo, C. Regazzoni, and A. Cavallaro, "Particle PHD filtering for multi-target visual tracking," in *Proc. Int. Conf. Acoustics, Speech and Signal Proc.*, (Honolulu, Hawaii, U.S.A.), Apr. 2007.

[116] R. Hoseinnezhad, B. Vo, B. Vo, and D. Suter, "Visual tracking of numerous targets via multi-Bernoulli filtering of image data," *Pattern Recognition*, vol. 45, pp. 3625–3635, Oct. 2012.

[117] R. Hoseinnezhad, B. Vo, and B. Vo, "Visual tracking in background subtracted image sequences via multi-Bernoulli filtering," *IEEE Trans. Signal Proc.*, vol. 61, pp. 392–397, Jan. 2013.

[118] J. L. Williams, "Hybrid Poisson and multi-Bernoulli filters," in *Proc. Int. Conf. Information Fusion*, (Singapore), Jul. 2012.

[119] J. L. Williams, "Marginal multi-Bernoulli filters: RFS derivation of MHT, JIPDA and association-based MeMBer," *IEEE Trans. Aerospace and Elec. Sys.*, To appear.

[120] E. Pollard, B. Pannetier, and M. Rombaut, "Hybrid algorithms for multitarget tracking using MHT and GM-CPHD," *IEEE Trans. Aerospace and Elec. Sys.*, vol. 47, pp. 832–847, Apr. 2011.

[121] K. Panta, B.-N. Vo, and S. Singh, "Novel data association schemes for the probability hypothesis density filter," *IEEE Trans. Aerospace and Elec. Sys.*, vol. 43, pp. 556–570, Apr. 2007.

[122] E. Delande, E. Duflos, D. Heurguier, and P. Vanheeghe, "Multi-target PHD filtering: proposition of extensions to the multi-sensor case," *Research Report RR-7337, INRIA*, Jul. 2010.

[123] P. Braca, S. Marano, V. Matta, and P. Willett, "Asymptotic efficiency of the PHD in multitarget/multisensor estimation," *IEEE J. Sel. Topics in Sig. Proc.*, vol. 7, pp. 553–564, Jun. 2013.

[124] E. Delande, E. Duflos, P. Vanheeghe, and D. Heurguier, "Multi-sensor PHD: Construction and implementation by space partitioning," in *Proc. Int. Conf. Acoustics, Speech and Signal Proc.*, (Prague, Czech Republic), May 2011.

[125] E. Delande, E. Duflos, P. Vanheeghe, and D. Heurguier, "Multi-sensor PHD by space partitioning: computation of a true reference density within the PHD framework," in *Proc. Stat. Signal Proc. Workshop*, (Nice, France), Jun. 2011.

[126] X. Jian, F.-M. Huang, and Z.-L. Huang, "The multi-sensor PHD filter: Analytic implementation via Gaussian mixture and effective binary partition," in *Proc. Int. Conf. Inf. Fusion*, (Istanbul, Turkey), Jul. 2013.

[127] S. Nagappa and D. E. Clark, "On the ordering of the sensors in the iterated-corrector probability hypothesis density (PHD) filter," in *Proc. SPIE Int. Conf. Sig. Proc., Sensor Fusion, Target Recog.*, (Orlando, FL, U.S.A.), Apr. 2011.

[128] C. Ouyang and H. Ji, "Scale unbalance problem in product multisensor PHD filter," *Electronics letters*, vol. 47, no. 22, pp. 1247–1249, 2011.

[129] B. F. La Scala and G. W. Pulford, "A Viterbi algorithm for data association," in *Proc. Int. Radar Symposium*, (Munich, Germany), Sep. 1998.

[130] G. W. Pulford, "Multi-target Viterbi data association," in *Proc. Int. Conf. Inf. Fusion*, (Florence, Italy), Jul. 2006.

[131] J. K. Wolf, A. M. Viterbi, and G. S. Dixon, "Finding the best set of K paths through a trellis with application to multitarget tracking," *IEEE Trans. Aerospace and Elec. Sys.*, vol. 25, pp. 287–296, Apr. 1989.

[132] U. Orguner, C. Lundquist, and K. Granstrom, "Extended target tracking with a cardinalized probability hypothesis density filter," in *Proc. Int. Conf. Inf. Fusion*, (Chicago,IL, U.S.A.), Jul. 2011.

[133] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas and Propagation*, vol. 34, pp. 276–280, Mar. 1986.

[134] R. Roy, A. Paulraj, and T. Kailath, "ESPRIT–A subspace rotation approach to estimation of parameters of cisoids in noise," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 34, pp. 1340–1342, Oct. 1986.

[135] B. D. Rao and K. Hari, "Performance analysis of root-MUSIC," in *Proc. Int. Conf. Acoustics, Speech and Signal Proc.*, (Glasgow, Scotland), May 1989.

[136] M. Viberg and B. Ottersten, "Sensor array processing based on subspace fitting," *IEEE Trans. Signal Processing*, vol. 39, pp. 1110–1121, May 1991.

[137] D. Angelosante, E. Biglieri, and M. Lops, "Multiuser detection in a dynamic environment - part II: Joint user identification and parameter estimation," *IEEE Trans. Information Theory*, vol. 55, pp. 2365–2374, May 2009.

[138] D. Angelosante, E. Biglieri, and M. Lops, "Sequential estimation of multipath MIMO-OFDM channels," *IEEE Trans. Sig. Proc.*, vol. 57, pp. 3167–3181, Aug. 2009.

[139] D. Angelosante, *Applications of random set theory to some problems in wireless communications*. PhD thesis, University of Cassino, Cassino, Italy, 2008.

[140] R. Mahler, "PHD filters for nonstandard targets, II: Unresolved targets," in *Proc. Int. Conf. Inf. Fusion*, (Seattle, WA, U.S.A.), Jul. 2009.

[141] M. Beard, B.-T. Vo, and B.-N. Vo, "Bayesian multi-target tracking with merged measurements using labelled random finite sets," *IEEE Trans. Signal Processing*, vol. 63, pp. 1433–1447, Mar. 2015.

[142] B. Vo, B. Vo, N. Pham, and D. Suter, "Joint detection and estimation of multiple objects from image observations," *IEEE Trans. Signal Proc.*, vol. 58, pp. 5129–5141, Oct. 2010.

[143] E. Ozkan, M. Guldogan, U. Orguner, and F. Gustafsson, "Ground multiple target tracking with a network of acoustic sensor arrays using PHD and CPHD filters," in *Proc. Int. Conf. Inf. Fusion*, (Chicago, IL, U.S.A.), Jul. 2011.

[144] R. Hoseinnezhad, B.-N. Vo, B.-T. Vo, and D. Suter, "Bayesian integration of audio and visual information for multi-target tracking using a CB-MeMBer filter," in *Proc. Int. Conf. Acoustics, Speech and Signal Proc.*, (Prague, Czech Republic), May 2011.

[145] P. Choppala, P. Teal, and M. Frean, "Adapting the multi-Bernoulli filter to phased array observations using MUSIC as pseudo-likelihood," in *Proc. Int. Conf. Information Fusion*, (Salamanca, Spain), Jul. 2014.

[146] P. Choppala, *Bayesian multiple target tracking*. PhD thesis, Victoria University of Wellington, Wellington, New Zealand, 2014.

[147] F. Papi and D. Y. Kim, "A particle multi-target tracker for superpositional measurements using labeled random finite sets," *ArXiv e-prints, http://arxiv.org/abs/1501.02248*, Dec. 2014.

[148] F. Papi, B.-N. Vo, B.-T. Vo, C. Fantacci, and M. Beard, "Generalized labeled multi-Bernoulli approximation of multi-object densities," *ArXiv e-prints, http://arxiv.org/abs/1412.5294*, Dec. 2014.

[149] S. Chib and E. Greenberg, "Understanding the Metropolis-Hastings algorithm," *The American Statistician*, vol. 49, pp. 327–335, Nov. 1995.

[150] G. Casella and E. I. George, "Explaining the Gibbs sampler," *The American Statistician*, vol. 46, pp. 167–174, Aug. 1992.

[151] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan, "Interacting multiple model methods in target tracking: a survey," *IEEE Trans. Aerospace and Elec. Sys.*, vol. 34, pp. 103–123, Jan 1998.

[152] A. Doucet, N. Gordon, and V. Krishnamurthy, "Particle filters for state estimation of jump Markov linear systems," *IEEE Trans. Signal Proc.*, vol. 49, pp. 613–624, Mar. 2001.

[153] F. Thouin, S. Nannuru, and M. Coates, "Multi-target tracking for measurement models with additive contributions," *http://networks.ece.mcgill.ca/node/189*.

[154] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *J. Comp. and App. Mathematics*, vol. 20, pp. 53–65, Nov. 1987.

[155] G. Schwarz, "Estimating the dimension of a model," *Annals of statistics*, vol. 6, pp. 461–464, Mar. 1978.

[156] R. Thorndike, "Who belongs in the family?," *Psychometrika*, vol. 18, pp. 267–276, Dec. 1953.

[157] F. Thouin, *Bayesian Inference in Networks*. PhD thesis, McGill University, Montreal, QC, Canada, 2011.

[158] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. U.S.A.: John Wiley & Sons, 2004.

[159] P. Djuric, T. Lu, and M. Bugallo, "Multiple particle filtering," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Proc.*, (Honolulu, HW, U.S.A.), Jun. 2007.

[160] M. Garey and D. Johnson, "Computers and intractability: A guide to the theory of NP-completeness," *WH Freeman & Co., San Francisco*, 1979.

[161] D. Knuth, "Dancing links," in *Millennial Perspectives in Computer Science* (J. W. J. Davies, B. Roscoe, ed.), pp. 187–214, Palgrave Macmillan, Basingstoke, 2000. arXiv preprint cs/0011047.

[162] J. Y. Yu, D. Ustebay, S. Blouin, M. Rabbat, and M. Coates, "Distributed underwater acoustic source localization and tracking," in *Proc. Asilomar Conf. Sig., Sys. and Computers*, (Pacific Grove, CA, U.S.A.), Nov. 2013.