Real-Time Sensor Management Strategies for Multi-Object Tracking

Marcos Eduardo Gomes Borges

N° d'ordre : 369

Centrale Lille

THESE

Présentée en vue
d'obtenir le grade de

DOCTEUR

en

Spécialité : Automatique, Génie Informatique,
Traitement du Signal et Image

par

Marcos Eduardo Gomes-Borges

Doctorat delivré par Central Lille

Real-Time Sensor Management Strategies for Multi-Object Tracking

Détermination et implémentation temps-réel de stratégies de gestion de capteurs pour le pistage multi-cibles

Soutenue le 19 décembre 2018 devant le jury d'examen :

Président	François Septier	Professeur, Université Bretagne Sud
Rapporteur	Eric Chaumette	Professeur, Institut Supérieur de l'Aéronautique et de l'Espace
Rapporteur	Lyudmila Mihaylova	Professeur, University of Sheffield (UK)
Examinateur	Audrey Giremus	Maitre de Conférence, Université Bordeaux 1
Directeur de thèse	Philippe Vanheeghe	Professeur, Centrale Lille
Co-directeur de thèse	Emmanuel Duflos	Professeur, Centrale Lille
Encadrant de thèse	Dominique Maltese	Ingénieur, Safran Electronics & Defense

Préparée au Centre de Recherche en Informatique, Signal et Automatique de Lille
CRIStAL UMR CNRS 9189

Ecole Doctorale SPI 072 (Lille I, Lille III, Artois, ULCO, UVHC, Centrale Lille)

Abstract

Modern surveillance systems often employ multiple controllable sensors that can collect information about objects of interest in their field of view. These systems must coordinate their observation strategies to enhance the information obtained by their future measurements in order to accurately estimate the states of objects of interest (location, velocity, appearance, etc). Therefore, adaptive sensor management consists of determining sensor measurement strategies that exploit *a priori* information in order to determine current sensing actions. One of the most challenging applications of sensor management is the multi-object tracking, which refers to the problem of jointly estimating the number of objects and their states or trajectories from noisy sensor measurements.

This thesis focuses on real-time sensor management strategies formulated in the Partially Observable Markov Decision Process (POMDP) framework to address the multi-object tracking problem within the Labeled Random Finite Set (LRFS) approach. The first key contribution is the rigorous theoretical formulation of the mono-sensor Labeled Probability Hypothesis Density (LPHD) filter with its Gaussian-mixture implementation. The second contribution is the extension of the mono-sensor LPHD filter for superpositional sensors, resulting in the theoretical formulation of the multi-sensor LPHD filter. The third contribution is the development of the Expected Risk Reduction (ERR) sensor management method based on the minimization of the Bayes risk and formulated in the POMDP and LRFS framework. Additionally, analyses and simulations of the existing sensor management approaches for multi-object tracking, such as Task-based, Information-theoretic, and Risk-based sensor management, are provided.

For the needs of both this and future research, a toolbox for sensor management and multi-object tracking has been developed using Matlab. It features many filtering methods for multi-object tracking, including the well-known PHD filters, the Generalized Labeled Multi-Bernoulli (GLMB) filters, and the new mono- and multi-sensor LPHD filters. Also included in the toolbox are the Kalman filter, the Extended Kalman filter (EKF), the Unscented Kalman filter (UKF), and the Particle filters. In addition, the toolbox provides the implementation of sensor management algorithms such as the Posterior Expected Number of Targets (PENT), Balanced Explorer and Tracker (BET), Posterior Expected Error of Cardinality and States (PEECS), the Mission-oriented and Risk-based sensor management, as well as the Rényi, Kullback-Leibler, and Cauchy-Schwarz divergences. Numerous examples are provided to demonstrate the usage of each algorithm, making the toolbox useful for building proof-of-concept implementations of multi-object tracking and sensor management algorithms. The latest version can be found at www.marcosgomesborges.com.

Acknowledgements

First, I would like to warmly thank my PhD advisor, Dominique Maltese, for offering me the opportunity to work on his fascinating multi-object tracking project. Thank you for everything I have learned with you, and for your patience and great sense of humor. Your flexibility gave me the freedom to study many interesting problems, making these last few years an extensive, enriching learning experience for me. Working with you was a delightful experience that I shall never forget.

I am very grateful to Philippe Vanheeghe and Emmanuel Duflos, my PhD directors, for their supervision and commitment, which kept me motivated even during the hard times of the thesis.

I would like to thank Prof. Eric Chaumette and Prof. Lyudmila Mihaylova for their real interest to this thesis through their excellent and impressive reviews. I am grateful to them and to the jury president Prof. François Septier, the examiner Prof. Audrey Giremus as well, for their questions and the discussions during the day of my defense.

I would also like to express my gratitude to my many wonderful colleagues for their support and positive attitudes, which contributed to brightening my time at Safran Electronics & Defense and École Centrale de Lille by making them places of interesting conversations and work.

I would like to acknowledge the Brazilian National Council for Scientific and Technological Development (CNPq), the French National Association for Research and Technology (ANRT), and Safran for funding the research reported in this thesis. At the ANRT, thanks to Clarisse Angelier and Christine Calore for their administrative support and their excellent work in partnership with French and Brazilian companies and academic institutions.

I am very grateful to Murilo Vicentin and Karolina Jasinska for being such incredible friends. I have had the privilege of knowing you from the beginning of my life in France and I feel so blessed for that.

Special thanks to Kasia Brodowska-Klinger for all your love and support, and for inspiring me to do something incredible. You helped me see what I could be. You have no idea how much happiness you brought into my life. Bardzo Cię kocham!

Finally, my deepest gratitude goes to my family, without whom this work would not have been possible. Thank you for always supporting me.

Eu gostaria de agradecer a toda minha família, em especial, Fernando, Juliana, mamãe e minha vó-mãe, sem a qual eu jamais seria a pessoa que sou hoje.

Contents

List of Acronyms

δ-GLMB	δ-Generalized Labeled Multi-Bernoulli
ADAS	Advanced Driver Assistance Systems
BET	Balanced Explorer and Tracker
CPHD	Cardinalized PHD
EKF	Extended Kalman filter
ERR	Expected Risk Reduction
FISST	Finite Set Statistics
FOV	field of view
GLMB	Generalized Labeled Multi-Bernoulli
GM	Gaussian-mixture
GM-LPHD	Gaussian-Mixture Labeled PHD
GM-PHD	Gaussian-Mixture PHD
GM-PHDT	Gaussian-Mixture PHD Tracker
i.i.d.	independent and identically distributed
JPDAF	Joint Probabilistic Data Association filter
KF	Kalman filter
KL	Kullback-Leibler divergence
l.i.i.d.	labeled independent and identically distributed
LMB	Labeled Multi-Bernoulli
LOSPA	Labeled Optimal SubPattern Assignment
LPHD	Labeled Probability Hypothesis Density
LRFS	Labeled Random Finite Set
MAP	Maximum a posteriori
MCMC	Markov Chain Monte Carlo
MDP	Markov Decision Process
MHT	Multiple Hypothesis Tracking

MMSE	Minimum Mean-Square Error
OMAT	Optimal Mass Transfer
OSPA	Optimal SubPattern Assignment
PDA	Probabilistic Data Association
PEECS	Posterior Expected Error of Cardinality and States
PENT	Posterior Expected Number of Targets
PENTI	Posterior Expected Number of Targets of Interest
PF	Particle filters
PGFl	Probability Generating Functional
PHD	Probability Hypothesis Density
PHDT	PHD Tracker
PIMS	Predicted Ideal Measurement Set
PMHT	Probabilistic Multiple Hypothesis Tracking
POMDP	Partially Observable Markov Decision Process
RFS	Random Finite Set
RMSE	Root Mean Square Error
SLAM	Simultaneous Localization and Mapping
SMC	Sequential Monte Carlo
SMC-PHD	Sequential Monte Carlo PHD
UAV	Unmanned Aerial Vehicles
UKF	Unscented Kalman filter

List of Symbols

k	time k
x	vector or scalar value
$\boldsymbol{x}_k$	labeled state vector at time k
x_k	state vector at time k
z_k	measurement vector at time k
p_{k-1}	posterior density at time $k-1$
p_k	posterior density at time k
$p_{k\|k-1}$	predicted density at time k given a posterior density at time $k-1$
p_+	predicted density (simplified form of $p_{k\|k-1}$)
$f_{k\|k-1}$	Markov transition density from time $k-1$ to time k
g_k	measurement likelihood at time k
π	multi-object probability density
$\boldsymbol{\pi}$	labeled multi-object probability density
ρ_n	cardinality distribution
$c(z)$	probability density of clutter measurements
$\kappa(z)$	probability hypothesis density of the clutter process
$\nu(x)$	probability hypothesis density (equivalent to intensity density)
$\mathcal{N}(x;\mu,\sigma)$	normal distribution over x with mean μ and standard deviation σ
X	set of unlabeled state vectors
$\boldsymbol{X}$	set of labeled state vectors
Z_k	set of measurements at time k
$Z_{1:k}$	set of all measurements up to time k
I	set of track labels (whithin a hypothesis)
L	set of track labels
$\emptyset$	empty set
$\mathbb{X}$	state space
$\mathbb{Z}$	measurement space
$\mathbb{L}$	label space (discrete)
$\mathbb{B}$	newborn labeled space
$\mathbb{C}$	component space (discrete)
$\mathbb{N}$	space of non-negative integers
$\mathcal{F}(\mathbb{X})$	finite subsets of the space $\mathbb{X}$

$\mathcal{F}_n(\mathbb{X})$	finite subsets of the space $\mathbb{X}$ with cardinality n
$\lvert X \rvert$	cardinality of a random finite set X
$\mathcal{L}(\cdot)$	label set extraction function (projection from $\mathbb{X} \times \mathbb{L}$ to $\mathbb{L}$)
$\Delta(\cdot)$	distinct label indicator function
$\delta_X(\cdot)$	generalized Kronecker delta function
$1_X(\cdot)$	inclusion function
$\langle f, g \rangle$	inner product of two functions f and g
h^X	multi-object exponential $\left(\prod_{x \in X} h(x) \right)$
θ	association hypothesis, e.g. track label to measurement
$\mathbb{E}[\cdot]$	expectation operator

Résumé

Cette thèse porte sur la définition de stratégies de gestion des capteurs en temps réel afin de résoudre le problème du suivi multi-objets dans le cadre d'ensemble aléatoire fini labéllisé ou « Labeled Random Finite Set (LRFS) ». Les quatre sections de ce résumé suivent les quatre chapitres de la thèse. La première section présente la base théorique de l'estimation Bayésienne pour les algorithmes de pistage mono et multi-objets dans le formalisme des ensembles aléatoires finis (RFS) standard et aussi labellisé. La deuxième section présente la formulation théorique du filtre mono-capteur LPHD « Labeled Probability Hypothesis Density » avec son implémentation sous l'hypothèse Gaussienne. La troisième section définit en détail l'extension du filtre LPHD pour le cas multi-capteurs. La quatrième section présente le développement des méthodes de gestion de capteurs formulés dans les cadres LRFS et POMDP « Partially Observable Markov Decision Process », suivi de la conclusion et travaux futurs.

Principes fondamentaux du pistage multi-objets

Le suivi (ou pistage) d'objets est l'opération consistant à estimer l'état des objets à travers le temps. Cette opération repose essentiellement sur des techniques de filtrage largement développées dans le cadre Bayésien. Le point difficile du pistage multi-cibles est l'extraction optimale d'informations utiles sur l'état de la cible à partir d'observations bruitées [BP99; BWT11]. En général, les observations ne décrivent pas exactement les positions des différentes cibles. Il est aussi possible, qu'elles ne correspondent à aucune cible (clutter ou fausses alarmes). En général, le nombre de cibles observées est inconnu et variable au cours du temps (les cibles peuvent apparaitre et disparaitre). Il faut donc mettre au point des outils capable d'estimer correctement le nombre de cibles et d'évaluer le mieux possible les paramètres des cibles présentes, en ne disposant que d'observations et d'un modèle statistique définis.

Le pistage multi-cibles est généralement réalisé en gardant l'estimation de l'état de la cible au fil du temps en utilisant des algorithmes tels que les algorithmes : JPDA « Joint Probabilistic Data Association » et MHT « Multiple Hypothesis Tracking » [BWT11]. La principale caractéristique de ces approches est la stratégie appliquée pour résoudre le problème de l'association des données, ce qui entraîne malheureusement une estimation Bayésienne non optimale [Mah14].

Plus récemment, Mahler a proposé une formulation alternative qui évite les associations explicites entre les mesures et les cibles. Cette nouvelle approche met en œuvre la théorie des ensembles aléatoires finis, RFS « Random Finite Set », afin de résoudre le problème du pistage multi-cibles dans un cadre Bayésien [Mah03a; Mah14]. Les algorithmes basés sur l'approche RFS consistent à propager au cours du temps une densité de probabilité π décrivant le nombre de cibles et leurs états en fonction de l'arrivée des nouvelles mesures

utilisant le schéma du filtre Bayésien multi-objets comme illustré par le schéma suivant:

$$\cdots \rightarrow \pi_{k-1}(X_{k-1}|Z_{1:k-1}) \xrightarrow{\text{prédiction}} \pi_{k|k-1}(X_k|Z_{1:k-1}) \xrightarrow{\text{mise à jour}} \pi_{k|k}(X_k|Z_{1:k}) \rightarrow \cdots$$

L'objectif de l'estimateur Bayésien multi-objets est de déterminer la densité a posteriori de l'état multi-objets $\pi_k(X_k|Z_{1:k})$ à l'instant k, où $Z_{1:k} = \{Z_1, \cdots, Z_k\}$ représente la séquence d'observations jusqu'à l'instant k. Ce filtre nécessite de deux distributions *à priori* : la densité de transition multi-objets Markovienne $f_{k|k-1}(X_k|X_{k-1})$ et la fonction de vraisemblance multi-objets $g_k(Z_k|X_k)$. La densité a posteriori multi-objets peut être calculée de manière récursive via les étapes de « prédiction » et de « mise à jour » du filtre comme décrit ci-dessous :

$$\pi_{k|k-1}(X_k|Z_{1:k-1}) = \int f_{k|k-1}(X_k|X_{k-1})\pi_{k-1}(X_{k-1}|Z_{1:k-1})\delta X_{k-1} \tag{1}$$

$$\pi_{k|k}(X_k|Z_{1:k}) = \frac{g_k(Z_k|X_k)\pi_{k|k-1}(X_k|Z_{1:k-1})}{\int g_k(Z_k|X_k)\pi_{k|k-1}(X_k|Z_{1:k-1})\delta X_k} \tag{2}$$

Toutefois, le filtre Bayésien multi-cibles utilisant les RFS n'a pas de solution explicite, il est donc nécessaire d'utiliser des approximations. Les plus connues d'entre elles sont les méthodes : PHD « Probability Hypothesis Density » [Mah03b], CPHD « Cardinalized PHD » [Mah07a] et GLMB « Generalized Labeled Multi-Bernoulli » [VV13; VVP14].

Le filtre PHD est une approximation du filtre Bayésien multi-cibles qui restreint les densités de probabilité $\pi_{k|k-1}(X_k|Z_{1:k-1})$ et $\pi_{k|k}(X_k|Z_{1:k})$ par leurs densités des moments du premier ordre notées $\nu_{k|k-1}(X_k|Z_{1:k-1})$ et $\nu_{k|k}(X_k|Z_{1:k})$, qui se propagent dans le temps comme illustré par le schéma suivant :

$$\begin{array}{ccccccc}
\cdots \rightarrow & \pi_{k-1}(X_{k-1}|Z_{1:k-1}) & \xrightarrow{\text{prédiction}} & \pi_{k|k-1}(X_k|Z_{1:k-1}) & \xrightarrow{\text{mise à jour}} & \pi_{k|k}(X_k|Z_{1:k}) \rightarrow \cdots \\
& \downarrow & & \downarrow & & \downarrow \\
\cdots \rightarrow & \nu_{k-1}(x_{k-1}|Z_{1:k-1}) & \xrightarrow{\text{prédiction}} & \nu_{k|k-1}(x_k|Z_{1:k-1}) & \xrightarrow{\text{mise à jour}} & \nu_{k|k}(x_k|Z_{1:k}) \rightarrow \cdots
\end{array}$$

Le filtre PHD peut être formulé de manière récursive via les étapes de « prédiction » et de « mise à jour » comme décrit ci-dessous [Mah03a; Mah07b; Mah14] :

$$\nu_{k|k-1}(x) = \gamma_k(x_k) + \int \Big(P_{S,k}(x_{k-1})f_{k|k-1}(x_k|x_{k-1}) + \beta_{k|k-1}(x|x_{k-1})\Big)\nu_{k-1}(x_{k-1})\mathrm{d}x_{k-1} \tag{3}$$

$$\nu_{k|k}(x_k) = \Big(1 - P_{D,k}(x_k)\Big)\nu_{k|k-1}(x_k) + \sum_{z\in Z_k} \frac{P_{D,k}(x_k)g_k(z|x_k)\nu_{k|k-1}(x_k)}{\kappa_k(z) + \int P_{D,k}(x_k)g_k(z|x_k)\nu_{k|k-1}(x_k)\mathrm{d}x_k} \tag{4}$$

où ν_{k-1} est la densité PHD à l'instant $k-1$, $\gamma_k(\cdot)$ est l'intensité de naissance de cibles à l'instant k, $P_{S,k}(x_{k-1})$ la probabilité d'existence d'une cible à l'instant k, $f_{k|k-1}$ la densité de la transition Markovienne, $\beta_{k|k-1}(x|x_{k-1})$ l'intensité d'un nouvel objet apparaissant basé sur l'état précédent x_{k-1} d'un objet déjà existant, $P_{D,k}(x_k)$ la probabilité qu'un objet avec l'état x_k soit détecté, et Z_k est l'ensemble d'observations à l'instant k.

Le filtre LPHD mono capteur

Les premiers filtres basés sur l'approche RFS ne peuvent générer que l'estimations de l'état (filtrage multi-objets), et ces algorithmes ne prennent pas en compte l'estimations de trajectoires (suivi multi-objets) en raison de la complexité de calcul. Certaines solutions proposées pour résoudre ce problème étaient généralement basées sur des méthodes heuristiques. La théorie RFS labellisée de Ba-Tuong Vo et Ba-Ngu Vo a été la première formulation théorique rigoureuse pour résoudre le problème du suivi multi-objet [VV13; Mah14; VVP14]. Leur proposition a abouti au filtre GLMB « Generalized Labeled Multi-Bernoulli », qui est le premier algorithme reconnu comme une solution Bayésienne optimale pour le suivi multi-objets.

La différence entre le RFS standard et le RFS labellisée réside dans le fait que les filtres RFS standard ne produisent que de multiples estimations d'état (filtrage multi-objets), tandis que les filtres LRFS produisent des estimations de trajectoires (suivi multi-objets). La version originale du filtre GLMB est quasiment inexploitable, et de nombreuses approximations ont été proposées. Ba-Ngu Vo, Ba-Tuong et Hoang ont proposé une implémentation rapide du filtre GLMB avec une prédiction et une mise à jour conjointes présentant une complexité de calcul en $O(n^2m)$, où n est le nombre d'objets et m est le nombre de mesures [HVV15; VVH17]. Cette mise en œuvre efficace a un temps d'exécution moyen au moins deux fois moindre que la version d'origine.

Dans le Chapitre 2, nous nous attachons à répondre aux questions suivantes : s'il existe-t-il une version du filtre PHD labellisée (LPHD), quelle est sa relation avec le filtre GLMB ? Est-ce une alternative sous-optimale mais plus rapide en calcul ? Puis nous montrons qu'une approximation du filtre GLMB, connue sous le nom de filtre LMB « Labeled Multi-Bernoulli » [Reu+14], peut être réinterprétée comme un filtre LPHD suivant le schéma ci-dessous :

$$\cdots \rightarrow \quad \nu_{k-1}(x,\ell) \quad \xrightarrow{\text{prédiction}} \quad \nu_{k|k-1}(x,\ell) \quad \xrightarrow{\text{mise à jour}} \quad \nu_{k|k}(x,\ell) \rightarrow \cdots$$

et selon Mahler [Mah14; Mah17]:

$$\nu_{k-1}(x,\ell) = r_{k-1}^{(\ell)} p_{k-1}(x,\ell) = r_{k-1}^{(\ell)} p_{k-1}^{(\ell)}(x) \tag{5}$$

L'équation 5 représente un LPHD, où ù $0 \leq r_{k-1}^{(\ell)} \leq 1$ et $p_{k-1}^{(\ell)}(x)$ représentent la probabilité d'existence et la distribution spatiale de l'objet avec le label ℓ. L'équation de prédiction pour le filtre LPHD est la même que pour le filtre LMB. Ainsi, l'étape de prédiction du filtre LPHD peut être exprimée par :

$$\nu_+(x,\ell) = \begin{cases} \int p_S(x',\ell)\, f_+(x|x')\, \nu(x',\ell)\mathrm{d}x & \text{if } \ell \in \mathbb{L} \\ \nu^B(x,\ell) & \text{if } \ell \in \mathbb{B} \end{cases} \tag{6}$$

où $\mathbb{B} \subseteq \mathbb{L}_+$ est l'ensemble labellisé pour les nouveaux cibles (naissance de cibles), $\mathbb{L}$ est l'ensemble de label pour les cibles actuelles, et $\nu^B(x,\ell) = r_B^{(\ell)} p_B^{(\ell)}(x,\ell)$ est la LPHD pour les nouvelles cibles.

Les équations de mise à jour pour le filtre LPHD sont les mêmes que pour le filtre LMB. Ainsi, l'étape de mise à jour du filtre exprimée par :

$$\nu(x', \ell') = L_Z(x', \ell') p_+(x', \ell') \tag{7}$$

où $L_Z(x', \ell')$ est le modèle d'observation qui dépend de l'application.

La théorie du LRFS fournit la base théorique pour une solution Bayésienne optimale pour le suivi multi-objets. Contrairement aux solutions heuristiques du filtre PHD avec label, le filtre LPHD proposé implémente complètement la théorie RFS labellisé. Cependant, la complexité de calcul de la mise à jour du filtre LPHD est nettement supérieure à celle des autres implémentations du filtre PHD utilisant le RFS standard.

Le filtre LPHD peut être considéré comme une approximation du filtre δ-GLMB, qui approxime la densité a posteriori multi-objet à l'aide d'un RFS LMB afin de simplifier l'étape de prédiction. Cette approximation résulte en un algorithme de suivi de cible précis et en temps réel. De plus, le filtre LPHD facilite la mise en œuvre à l'aide de mélanges gaussiens et n'a pas tendance à se dégrader lorsque le clutter est dense ou que les objets sont proches.

Le filtre LPHD multi-capteurs

Le filtre mono-capteur LPHD ne peut être utilisé qu'avec le modèle de mesure multi-objets standard, ce qui suppose qu'un objet génère au plus une seule mesure (une détection), que toute mesure est générée par au plus un seul objet, et que les mesures non générées par des objets sont de fausses détections ou du clutter. En revanche, les capteurs tels que les radars et les sonars ne respectent pas le modèle de mesure standard car leurs mesures sont composées d'une somme des signaux observé.

Les systèmes multi-capteurs et les capteurs qui ne suivent pas le modèle de mesure standard sont partout, et les filtres multi-objets spécialement conçus pour ceux-ci pourraient apporter une amélioration significative par rapport aux approches de suivi classiques à détection standard mono-capteurs. En 2009, Mahler a présenté la formulation théorique du filtre PHD multi-capteurs non labellisé pour le cas avec deux capteurs [Mah09b; Mah09c]. Par la suite, Delande a introduit un développement rigoureux du filtre PHD non labellisé pour le cas multicapteurs en 2010 [Del+10; Del12], et Liu a proposé une autre extension multicapteurs non labellisée limitée aux systèmes de capteurs linéaires en 2011 [LW11].

Cette section résume le développement du filtre LPHD multicapteurs en tant qu'extension du filtre LPHD monocapteur basé sur les travaux de Papi, Saucan et Mahler et al. [PK15; SLC17; Mah18]. Le filtre PHD multi-capteurs proposé diffère des filtres mentionnés ci-dessus en ce sens que son développement est basé sur le formalisme du LRFS et la distribution RFS LMB, plutôt que d'utiliser la distribution du formalisme RFS standard et distribution Poisson.

Un modèle de mesure standard peut être exprimé par :

$$Z_k = \eta_k(x) + V_k \tag{8}$$

où x est l'état d'un objet, $\eta_k(x)$ c'est une fonction de mesure non linéaire, et V_k est un vecteur

de bruit aléatoire de moyenne nulle. La fonction de vraisemblance du capteur de ce modèle est définie par :

$$L_z(x) = g_k(z|x) = g_{V_k}\Big(z - \eta_k(x)\Big) \tag{9}$$

Pour les capteurs comme les radars et les sonars qui n'obéissent pas au modèle de mesure standard, le modèle de mesure est donné par :

$$Z_k = \eta_k(X) + V_k = \eta_k(x_1) + \cdots + \eta_k(x_n) + V_k \tag{10}$$

où $X = \{x_1, \cdots, x_n\}$ with $|X| = n$ est l'ensemble des états multi-objets. En général, $\eta_k(x)$ et V_k sont des vecteurs. La densité de mesure de ce modèle est :

$$\begin{aligned} L_z(X) &= g_k(z|X) \\ &= g_{V_k}(z - \eta_k(X)) \\ &= g_{V_k}(z - \eta_k(x_1)) - \cdots - g_{V_k}(z - \eta_k(x_n)) \end{aligned} \tag{11}$$

L'équation de prédiction pour le filtre LPHD multi-capteurs est la même que celle du filtre LPHD mono-capteur et est donnée par :

$$\nu_{k|k-1}(x,\ell) = \begin{cases} \int p_S(x',\ell)\, f_{k|k-1}(x|x')\, \nu(x',\ell)\mathrm{d}x & \text{if } \ell \in \mathbb{L} \\ \nu^B(x,\ell) & \text{if } \ell \in \mathbb{B} \end{cases} \tag{12}$$

où $\mathbb{B} \subseteq \mathbb{L}_{k|k-1}$ est l'ensemble labellisé pour les nouvelles cibles et $\mathbb{L}$ est le label définie pour les cibles courantes.

La mise à jour LPHD multi-capteurs pour une prédiction LPHD $\nu_{k|k-1}(x,\ell)$ et de nouvelles mesures Z_k est définie par :

$$\nu_k(x,\ell) = \begin{cases} 0 & \text{if } \ell \notin J \\ \dfrac{\alpha_k\, \nu_{k|k-1}(x,\ell)\, \mathcal{N}(z; z_k - \eta(x,\ell) - o^{(\ell)}, R + O^{(\ell)})}{\max_{\ell' \in J} \int \nu_{k|k-1}(x',\ell')\, \mathcal{N}(z; z_k - \eta(x',\ell') - o^{(\ell')}, R + O^{(\ell')})\, \mathrm{d}x'} & \text{otherwise} \end{cases} \tag{13}$$

où $0 < \alpha_k \leq 1$ c'est une constante dépendante de l'application, et $o^{(\ell)}$ et $O^{(\ell)}$ sont définis par :

$$o^{(\ell)} = \sum_{\ell' \in J - \{\ell\}} \nu^{(\ell')}[\eta], \tag{14}$$

$$O^{(\ell)} = \sum_{\ell' \in J - \{\ell\}} \Big(\nu_{(\ell')}[\eta\eta^T] - \nu^{(\ell')}[\eta]\nu^{(\ell')}[\eta]^T\Big) \tag{15}$$

où ^{T} désigne la matrice transposée, et :

$$\nu^{(\ell)}[\eta] = \int \eta(x,\ell)\nu_{k|k-1}(x,\ell)\mathrm{d}x, \tag{16}$$

$$\nu^{(\ell)}[\eta\eta^T] = \int \eta(x,\ell)\eta(x,\ell)^T\nu_{k|k-1}(x,\ell)\mathrm{d}x, \tag{17}$$

et similaire au filtre LPHD mono-capteur, $\int \nu_{k|k-1}(x,\ell)\mathrm{d}x \leq 1$ pour chaque ℓ.

Le filtre multi-objets pour les modelés de mesure non standard est toujours d'une complexité calculatoire trop importante. Mahler a tenté de résoudre ce problème en 2009, en proposant une formulation du filtre CPHD pour les capteurs qui suivent un modèle de mesure non standard, suivi d'autres travaux de Nannuru et al. en 2012 et 2013 [ME12; NCM13]. En 2015 et 2017, une extension du filtre GLMB a été proposée par Papi et Saucan et al. pour le rendre capable de gérer des modèles de mesures labellisées [PK15; SLC17]. Mais les approches de Papi et de Saucan exigent des calculs complexes pour un grand nombre d'objets.

Cette section peut être considéré comme une continuation aux travaux mentionnés précédemment pour obtenir un filtre multi-capteurs pouvant être traité dans le contexte LRFS. Il est présenté le développement du filtre multi-capteurs LPHD pour les capteurs qui suivent le modèle de mesure non standard (équation 10). La formulation proposée est également applicable à des approximations telles que le filtre LMB car ce filtre est un cas particulier du GLMB RFS [Reu+14]. Les implémentations possibles du filtre LPHD multicapteurs peuvent être basées sur les travaux de Papi et Beard et al. [Pap+14; BVV15].

Gestion de capteurs

Dans un environnement où les réseaux de capteurs prennent une dimension de plus en plus importante, voire stratégique, la gestion des capteurs dans le but d'optimiser un critère opérationnel est d'un grand intérêt. En effet, un nombre croissant d'applications mettent en œuvre un réseau de capteurs pour acquérir de l'information dans un contexte donné. Bien, qu'ils ne soient pas indispensables, ces capteurs, grâce à des caractéristiques différentes, permettent d'utiliser leurs données de manière complémentaire, afin de disposer d'une information la plus riche possible. De plus, les capteurs disposent de plusieurs paramètres qui apportent un contrôle de l'information que l'on souhaite acquérir.

Les développements des technologies des capteurs ont conduit à l'émergence d'un grand nombre de degrés de liberté pour le contrôle des dispositifs de détection. La gestion de capteurs devient primordiale lorsque le système de détection devient capable de gérer activement ces ressources en réaction aux mesures précédentes. Le terme « gestion de capteur » fait référence au contrôle optimal d'un capteur agile (radar, lidar, camera PTZ, etc.) pour satisfaire les contraintes du système et atteindre des objectifs opérationnels [Her+08; HC11].

La gestion de capteurs multi-objets est un problème de contrôle optimal non linéaire. Non seulement le nombre de cibles varie au cours du temps, mais aussi les mesures peuvent varier en fonction de fausses alarmes, clutter ou non-détection de cibles. En ce qui concerne l'approche RFS, les principales méthodes sont : la gestion de capteurs basée sur les tâches [MZ04; Mah04; Del+10; Del12], le gain d'information [RV10; RVC11; Hoa+15], les risques et les menaces selon les objectifs de la mission [PR07; WHE11; Mar15; Gom+16; Gom+17].

La gestion de capteurs pour le pistage multi-cibles est d'une importance primordiale pour les systèmes de surveillance et de défense. Les cas d'utilisation proposées dans la thèse concernent trois domaines principaux : terrestre, aéroporté et maritime. L'objectif est de contrôler les capteurs de façon à détecter des événements, tels que des intrusions dans des zones interdites, le suivi des passages de véhicules sur un itinéraire et la détection d'actes de piraterie sur petites embarcations et leur neutralisation. Les cibles sont : des piétons, des véhicules, des mini-drones, des hélicoptères volant à basse altitude, et aussi des bateaux.

Cette thèse porte sur l'identification et le développement de stratégies de gestion de capteurs pour le suivi multi-objets pouvant être mises en œuvre en temps réel dans des situations réalistes. Nous utilisons l'approche RFS pour le suivi multi-objets afin de résoudre le problème de gestion multi-capteurs formulé dans le cadre du POMDP « Partially Observable Markov Decision Process ». Le Chapitre 4 présente les analyses et simulations des approches de gestion des capteurs existantes, telles que la gestion des capteurs basée sur la tâche, la théorie de l'information, la mission et la nouvelle méthode de gestion de capteur basée sur le risque Bayésien.

Ouvertures

De nombreuses pistes sont envisageables pour l'approfondissement des travaux conduits dans cette thèse. Le filtre LPHD fournit plusieurs sujets intéressants pour les recherches futures: en raison des propriétés LMB RFS, il est possible de calculer la variance du nombre de cibles de manière simple, en utilisant l'idée proposée par Schlangen et al. [Sch+18] pour créer un algorithme LPHD du deuxième ordre, qui peut faciliter l'intégration dans les plateformes de gestion de capteurs. Les techniques d'approximation peuvent être mises en œuvre pour réduire la complexité des filtres constituent un autre sujet de recherche.

Parmi les autres extensions possibles du filtre LPHD figurent l'implémentation basée sur les méthodes de Monte Carlo, la prise en charge de l'approche multi-modèle, l'application au SLAM et le développement d'un modèle de naissance des cibles différent. En plus, explorer différentes méthodes de « gating », « pruning » d'approximation peuvent être mises en œuvre pour réduire la complexité calculatoire des filtres LPHD mono-capteurs et multi-capteurs. Et pour la gestion de capteur, rechercher des solutions efficaces en termes de calcul en utilisant la nouvelle fonction coût pour la gestion de capteur : « Expected Risk Reduction (ERR) ». Coupler l'ERR au filtre LPHD avec un contrôle de capteur non myopique ainsi que la gestion de capteur dans une architecture de fusion de données distribuée.

Introduction

The progress in remote sensing technology led to the appearance of devices with more flexible configurations and set-up that can be updated in real time. Software command can update parameters such as center frequency, bandwidth, beam-form, beam-pointing, sampling rate, position, and many other aspects of sensors' operating modes. Sensor systems operate under resource constraints that prevent the constant simultaneous use of all resources; for a given sensing application, sensor management consists of determining sensing actions that maximize the efficiency of the resulting sensor measurements. Depending on the complexity of the system and the number of sensing actions available, optimal sensor management can be intractable. For most applications of interest, a large number of decisions must be made regarding how sensors should collect measurements, which makes sensor management challenging [BP99; XS02; Her+08; HC11; BWT11].

Sensor management is related to the control that an agile system has over the sensor's configuration to satisfy operational constraints and accomplish operational objectives [Her+08; HC11; BWT11]. Systems in which sensor management is currently used include autonomous robots, Advanced Driver Assistance Systems (ADAS), Unmanned Aerial Vehicles (UAV), surveillance, multi-function radars, and pan-tilt-zoom cameras, to name a few. Recently there have also been remarkable advances in networked systems as well as in autonomous and semi-autonomous vehicles, which are equipped with different types of sensors and interconnected by networks, leading to configurable networked sensing systems.

Multi-object tracking is one of the main applications of sensor management. Its objective is to estimate the number of objects and the state of their trajectories from measurements delivered by various types of sensors. The state is understood as a vector containing information about a target under surveillance, such as its position, velocity, and acceleration. Successive estimates provide the tracks that describe the trajectory of a target. Multi-object tracking is applied in several areas such as surveillance, defense, air traffic control, space-related problems, robotics, autonomous vehicles, collision avoidance, econometrics, oceanography, remote sensing, signal and image processing, and biomedical research [BP99; BLK01; BWT11; HC11; Mah14].

Motivation and Scope

The key to a successful multi-object tracking system lies in the optimal extraction of useful information about the target's state (position, velocity, acceleration, etc.) from the observations, despite the sensor's imperfections. This task is usually realized by maintaining an estimate of the target's state over time using algorithms such as the Joint Probabilistic Data Association filter (JPDAF) [Bar74], Multiple Hypothesis Tracking (MHT) [Rei79], and Random Finite Set (RFS) filters [Mah03a; Mah07b; Mah14]. The JPDAF and MHT approaches

are well established and they seem more relevant to the multi-object tracking systems used in industry, while the RFS approach is an emerging paradigm. JPDAF and MHT are formulated via data association followed by single-object filtering. Data association refers to the partitioning of the measurements into potential tracks and false alarms, whereas filtering is used to estimate the state of the target given its measurement history.

The main characteristic of these approaches is the strategy applied to solve the data association problem between measurements and targets, which unfortunately results in a non-Bayes-optimal estimation [Mah14]. More recently, an alternative formulation that avoids explicit associations between measurements and targets was proposed by Mahler [Mah03a]. This new approach uses the RFS theory in order to solve the multi-object tracking problem in a Bayesian framework. It offers a mathematically elegant representation of a finite but time-varying number of targets and measurements [Mah03a; Mah07b; Mah14].

Target tracking is essentially a dynamic state estimation or filtering problem for which the Bayesian approach provides the theoretical optimal filtering solution [Cla06; Pan07; Vo08; Pac11; Del12; BWT11; Mah14]. The multi-object Bayes filter is not tractable for real-time implementations due to the combinatorial complexity of the multiple target likelihoods. To overcome this limitation, approximations are required. A practical alternative to Bayesian multi-object tracking was proposed by Mahler to propagate the first-order statistical moment, or Probability Hypothesis Density (PHD), instead of the multi-object posterior itself [Mah03a]. In the mid 1990s, Mahler constructed Finite Set Statistics (FISST) from the mathematical theory of point processes and the random set theory as a way of extending classical mono-sensor single-object statistics to multi-sensor multi-object statistics. The multi-object states and observations are represented as an RFS from which a theoretically optimal Bayesian multi-sensor multi-object filter can be derived [Mah07b; Mah14].

Nowadays, there are several algorithms based on the RFS approach, such as the PHD filter [Mah03a], the Cardinalized PHD (CPHD) filter [Mah07a], the Generalized Labeled Multi-Bernoulli (GLMB) filter [VV13; VVP14; Mah14], and the Labeled Multi-Bernoulli (LMB) filter [Reu+14]. In addition, beginning in the 1990s, the growing interest in techniques and theories for sensor management introduced new elements that must be considered in order to improve the overall performance of the multi-object tracking systems. In many multi-object tracking applications, a sensor can be controlled by changing the position, orientation, or motion of the sensor platform or by operating it in a different mode, which may have a significant impact on the quality of the estimation performance of the target tracking system. Modern surveillance systems often employ multiple controllable sensors that are capable of collecting information on objects of interest in their field of view. These sensors must coordinate their observation strategies in order to enhance the information that will be collected by their future measurements to estimate the states of objects of interest [BP99; XS02; Her+08; HC11].

Sensor management for multi-object tracking is typically an optimal nonlinear control problem. Not only does the number of targets vary in time, but the measurements are susceptible to missed detections and false alarms. With reference to the RFS approach, Mahler developed theoretical foundations of the multi-object sensor management reward function related to the Posterior Expected Number of Targets (PENT) [MZ04] and the Posterior Expected Number of Targets of Interest (PENTI) [Mah04; Mah14]. Delande introduced a reward function called Balanced Explorer and Tracker (BET), which provides efficient sensor management in situations where the sensor's field of view cannot cover the whole state space at the same time [Del12]. Ristic and Vo proposed a reward function to sensor management using the Rényi divergence between the multi-object prior and multi-object posterior den-

sities [RV10; RVC11]. More recently, Hoang, Vo, and Mahler offered a new intuitive and tractable objective function based on the Cauchy-Schwarz information functional [Hoa+15].

The scope of this thesis is the identification, analysis, and development of sensor management strategies for multi-sensor multi-object tracking algorithms that can be implemented in real-time under realistic situations. For the needs of both this and future research, a toolbox for sensor management and multi-object tracking has been developed using Matlab. It features many filtering methods for multi-object tracking, including the well-known PHD filters, the Cardinalized PHD (CPHD) filters, the Generalized Labeled Multi-Bernoulli (GLMB) filter, the Labeled Multi-Bernoulli (LMB) filter, and the new mono- and multi-sensor LPHD algorithms. The toolbox also provides the implementation of all sensor management algorithms analyzed in this thesis such as the Posterior Expected Number of Targets (PENT), Posterior Expected Number of Targets of Interest (PENTI), and the proposed Risk-based sensor management, as well as the Rényi and Kullback-Leibler divergences.

This thesis is supported by the Brazilian National Council for Scientific and Technological Development (CNPq) and the French National Association for Research and Technology (ANRT) through France's System of Industrial Agreements for Training through Research, known as the CIFRE program. Simulations and results presented in this thesis were conducted through the partnership between Safran Electronics & Defense and École Centrale de Lille by the Research Center in Computer Science, Signal and Automatic Control of Lille (CRIStAL UMR CNRS 9189).

Key Contributions

This thesis focuses on the identification and development of sensor management strategies for multi-object tracking that can be implemented in real-time under realistic situations. We use the Random Finite Set (RFS) approach to multi-object tracking to address the multi-sensor management problem formulated in the Partially Observable Markov Decision Process (POMDP) framework. Rigorous analysis and simulations of existing sensor management approaches, such as Task-based, Information-theoretic, Mission-oriented, and the new Task-based sensor management for multi-object tracking is performed. Succinct summaries of the state-of-the-art multi-object tracking algorithms are also presented.

The major contributions established in this thesis are as follows:

- Rigorous theoretical formulation of the mono-sensor Labeled Probability Hypothesis Density (LPHD) filter to address the multi-object tracking problem with its GM implementation, the Gaussian-Mixture Labeled PHD (GM-LPHD) filter.
- Extension of the mono-sensor LPHD filter for superpositional sensors, resulting in the theoretical formulation of the multi-sensor LPHD filter to address the multi-sensor multi-object tracking problem.
- Development of the Expected Risk Reduction (ERR) sensor management method based on the minimization of the Bayes risk and formulated in the POMDP and RFS framework. The ERR metric is based on the expected cost of an incorrect decision about a target's classification and conditioned on the event of losing a target track, which allows for the combination of classification and kinematic uncertainty in the same metric.

Publications

The following are the previous publications relating to the research conducted for this thesis:

1. M. E. Gomes-Borges, D. Maltese, P. Vanheeghe, E. Duflos. "Risk-based Sensor Management using Random Finite Sets," Proceedings IEEE, 20th International Conference on Information Fusion, July 2017.
2. M. E. Gomes-Borges, D. Maltese, P. Vanheeghe, G. Sella, E. Duflos. "Sensor Management using Expected Risk Reduction approach," Proceedings IEEE, 19th International Conference on Information Fusion, July 2016.

Thesis Outline

The thesis is divided into four chapters:

Chapter 1 presents an overview of the Bayesian estimation for single and multi-object tracking algorithms and the Random Finite Set (RFS) and labeled RFS formalism for filtering and tracking. Numerous filtering methods such as the PHD filters, the Kalman filter, the Extended Kalman filter (EKF), the Unscented Kalman filter (UKF), and the Particle filters are presented in this chapter, as well as tracking methods such as the PHD Tracker (PHDT) and the Generalized Labeled Multi-Bernoulli (GLMB) filters.

Chapter 2 presents the development of the Labeled Probability Hypothesis Density (LPHD) filter based on Mahler's work [Mah07b; Mah14; Mah03a; Mah17]. The first section provides an overview of the heuristic labeled PHD. Then the Labeled Multi-Bernoulli (LMB) filter is introduced as the basis of the rigorous theoretical formulation of the labeled PHD filter. A Gaussian-mixture implementation is proposed and the implementation of the Gaussian-Mixture Labeled PHD (GM-LPHD) filter is described in detail. Finally, the main results obtained on simulated data are discussed.

Chapter 3 describes the development of the multi-sensor LPHD filter for superpositional sensors. The superpositional measurement model is introduced, followed by the rigorous theoretical development of the multi-sensor LPHD filter.

Chapter 4 discusses in detail the novel Risk-based sensor management scheme that combines the RFS theory and the POMDP framework in the form of the Expected Risk Reduction (ERR) approach. This chapter also presents multiple examples comparing the ERR approach with other state-of-the-art sensor management algorithms.

Finally, **Conclusions** summarizes the main contributions of this thesis and outlines possible future research directions.

Chapter 1

Fundamentals of Multi-Object Tracking

This chapter aims to provide a theoretical background of Bayesian estimations for single and multi-object tracking algorithms based on the Random Finite Set (RFS). The unlabeled and labeled RFS for multi-object trajectory estimations are also discussed.

Section 1.1 presents the theory of Bayesian filtering and its relevance to target tracking. The Kalman filter is derived using properties of Gaussian distributions for linear systems. The Extended Kalman filter (EKF), the Unscented Kalman filter (UKF), and the Particle filters are derived for nonlinear systems.

Section 1.2 provides the theoretical tools that are required for the comprehension of the RFS framework, the multi-object Bayesian filtering, and the Probability Hypothesis Density (PHD) filter, as well as its closed-form solution known as the Gaussian-Mixture PHD filter. The Optimal SubPattern Assignment (OSPA) metric is also presented.

Section 1.3 provides a summary of popular state-of-the-art multi-object tracking algorithms, such as the PHD Tracker and the Generalized Labeled Multi-Bernoulli (GLMB) filter. This section also introduces the Labeled Random Finite Set (LRFS) formalism and notations used in the development of the Labeled Probability Hypothesis Density (LPHD) filter.

1.1 Bayesian filtering

The Bayesian approach provides an optimal filtering solution for target state estimation. The objective is to construct the probability density function (pdf or density) of the state based on all available information, including the sequence of received observations. Estimation can be defined as the problem of determining an unknown state x from a noisy observation z. The state refers to the physical properties such as position, velocity and acceleration of an object under surveillance. Noise is unwanted or unknown information that may increase the level of uncertainty in the state estimation. The dynamic system is noisy, and evolves as a function of time. In the context of this research, the term filtering refers to the process of filtering out the noise in the observation in order to provide an optimal estimate of the state based on the observed measurements and the assumptions made about the dynamic system [BLK01; RAG04; Cha+11].

The objective is to construct the posterior density of the target state using the obser-

vation history from time 1 to current time k, $z_{1:k} = (z_1, \cdots, z_k)$. The *a posteriori* density characterizes all statistical uncertainty in the target state. Consequently, it can be used to make any inference about the estimated state and its statistical uncertainty. The objective is to find an estimate of a target state x_k given a sequence of observations $z_{1:k}$. Figure 1.1 illustrates a simplified schematic of a single-object state estimation.

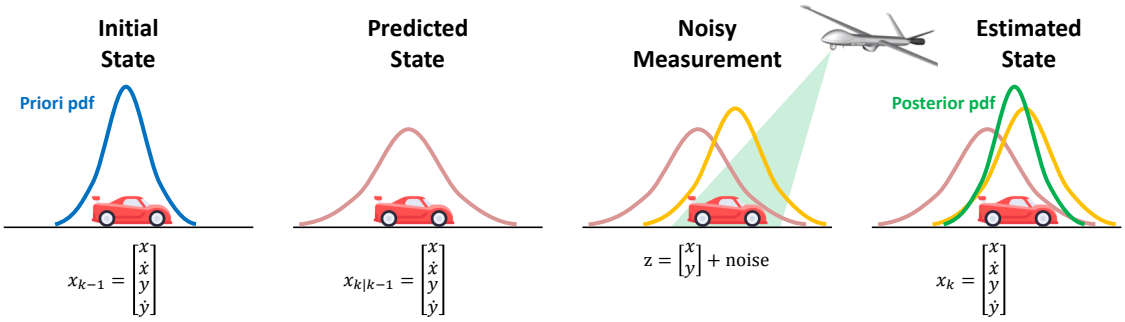

Figure 1.1: Simplified schematic of a vehicle state estimation. In this example, the blue Gaussian distribution represents the pdf, providing the initial belief in the estimate of the car state at time $k-1$. The red pdf shows that the confidence in the knowledge of the car state decreases during the prediction step due to the uncertainty associated with any process noise from accelerations or decelerations. The observation of the car state and the level of uncertainty in that noisy measurement are represented by the orange pdf. The green pdf is generated by multiplying the pdfs associated with the prediction and measurement of the car state at time k. This new pdf provides the best estimate of the car state by fusing the data from the prediction and the measurement.

The dynamic model of a target can be described by discrete- or continuous-time models [BP99; BLK01; Cha+11; BWT11]. This thesis only considers the discrete-time models. Hence, the target state x_k evolves in time according to the state transition equation:

$$x_k = \mathrm{f}_{k-1}(x_{k-1}, v_{k-1}) \tag{1.1}$$

where f_{k-1} is a known, possible nonlinear function that transforms any given state vector x_{k-1} and process noise v_{k-1} at time $k-1$ into a new state vector x_k at time k. In general, the state transition equation can be described in a probabilistic way by a Markov transition density:

$$f_{k|k-1}(x_k|x_{k-1}) \tag{1.2}$$

i.e., the probability density that a target with state vector x_{k-1} at time $k-1$ moves to the state x_k at time k. In such a system, with the so-called Markov-property, the state x_k given x_{k-1} is independent of the history of states and measurements. The objective of filtering is to recursively estimate x_k from observations z_k. The relationship between the observation and the target state is defined by the observation equation:

$$z_k = h_k(x_k, w_k) \tag{1.3}$$

where h_k is a known, possibly nonlinear function that transforms any given state vector x_k and observation noise w_k at time k into an observation vector z_k. Note that the process and

observation noise are assumed to be uncorrelated. The observation equation can be described in a probabilistic way by the likelihood function:

$$g_k(z_k|x_k) \tag{1.4}$$

i.e., the probability density that a target with state vector x_k generates an observation z_k. The observation z_k is independent of the history of observations and states. Figure 1.2 shows a schematic representation of the dynamic model.

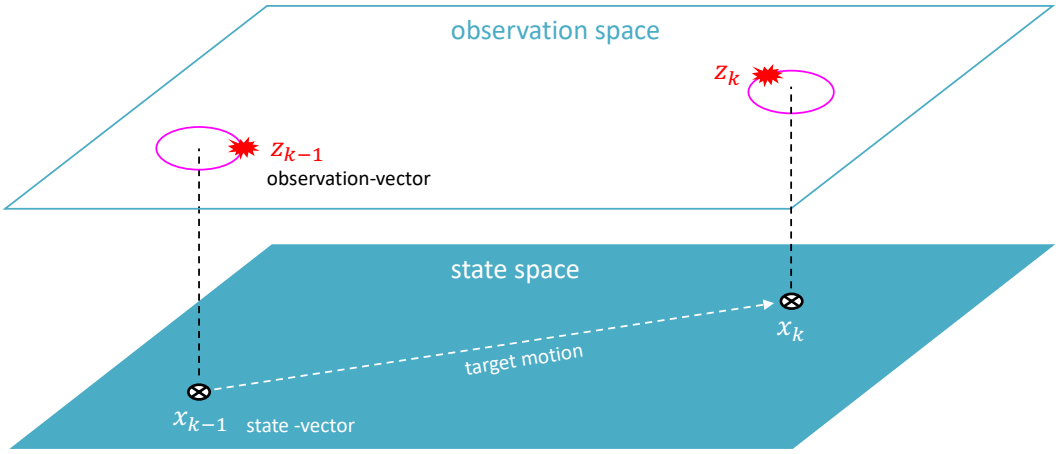

Figure 1.2: Simplified schematic of the dynamic model where a single-object state evolves in the state space. The objective is to estimate the state trajectory from the observation history.

1.1.1 The Bayes filter

The Bayes filter estimates the state vector x_k recursively in time based on the sequence of all available observations $z_{1:k}$ up to time k. The objective is to construct the posterior density $p_k(x_k|z_{1:k})$ of the state, which is calculated using two steps: prediction and update [BLK01; BWT11; Cha+11]. Supposing the *a priori* density $p_{k-1}(x_{k-1}|z_{1:k-1})$ at time $k-1$ is known. The prediction step involves the implementation of the dynamic model (1.2) to obtain the prediction density of the state at time k using the Chapman-Kolmogorov equation:

$$p_{k|k-1}(x_k|z_{1:k-1}) = \int f_{k|k-1}(x_k|x_{k-1}) p_{k-1}(x_{k-1}|z_{1:k-1}) \mathrm{d}x_{k-1} \tag{1.5}$$

Since a new measurement z_k is available at the current time k, the update step updates the predicted density obtained via (1.4) and (1.5) according to Bayes' rule:

$$p_{k|k}(x_k|z_{1:k}) = \frac{g_k(z_k|x_k) p_{k|k-1}(x_k|z_{1:k-1})}{\int g_k(z_k|x_k) p_{k|k-1}(x_k|z_{1:k-1}) \mathrm{d}x_k} \tag{1.6}$$

The recurrence relations (1.5) and (1.6) constitute the basis for the optimal Bayesian solution, i.e., a solution for exact and complete characterization of the posterior density in a recursive way. Knowledge of the *a posteriori* density allows for the computing of an optimal

state estimate concerning different criteria. For example, the Maximum a posteriori (MAP) estimate is the maximum of $p_{k|k}(x_k|z_{1:k})$ [BLK01; RAG04; BWT11; Cha+11]:

$$\hat{x}_{k|k}^{\text{MAP}} \overset{\text{def}}{=} \arg\max_{x_k} p_{k|k}(x_k|z_{1:k}) \tag{1.7}$$

while the Minimum Mean-Square Error (MMSE) estimate is the conditional mean of x_k:

$$\hat{x}_{k|k}^{\text{MMSE}} \overset{\text{def}}{=} \mathbb{E}[x_k|z_{1:k}] = \int x_k p_{k|k}(x_k|z_{1:k})\mathrm{d}x_k \tag{1.8}$$

In theory, the Bayes filter allows the construction of the exact posterior density $p_{k|k}(\cdot)$ recursively at each time. However, the Bayes filter recursion does not admit closed-form solutions due to computational intractability, and approximate solutions are often used. The Bayes filter recursion can be illustrated by the following diagram:

$$\cdots \rightarrow p_{k-1}(x_{k-1}|z_{1:k-1}) \xrightarrow{\text{prediction}} p_{k|k-1}(x_k|z_{1:k-1}) \xrightarrow{\text{update}} p_{k|k}(x_k|z_{1:k}) \rightarrow \cdots$$

1.1.2 The Kalman filter

When the dynamic and observation models are linear with additive Gaussian noise, the estimation problem given in the Bayesian filter has a closed-form solution known as the Kalman filter (KF) [Kál60; BP99; BLK01; RAG04; BJ05; BWT11; Cha+11; GA14]. It constitutes the basis for most of the more robust target tracking algorithms. The KF model assumes a linear discrete-time system given as follows [BLK01; LJ03]:

$$x_k = F_{k-1}x_{k-1} + v_{k-1} \tag{1.9}$$
$$z_k = H_k x_k + w_k \tag{1.10}$$

where F_{k-1} is a transition matrix of the dynamic model and H_k is the observation model matrix used to map state vector parameters into the measurement domain, while v_{k-1} and w_k are independent zero-mean Gaussian noise variables with covariance matrices Q_{k-1} and R_k, respectively. Hence, the transition density and likelihood function can also be expressed in probabilistic terms:

$$f_{k|k-1}(x_k|x_{k-1}) = \mathcal{N}(x_k; F_{k-1}x_{k-1}, Q_{k-1}) \tag{1.11}$$
$$g_k(z_k|x_k) = \mathcal{N}(z_k; H_k x_k, R_k) \tag{1.12}$$

where $\mathcal{N}(\cdot\,; m, P)$ denotes a Gaussian density with mean m and covariance P, and has the form:

$$\mathcal{N}(x; m, P) \overset{\text{def}}{=} \frac{1}{\sqrt{|2\pi P|}} \exp\left(-\frac{1}{2}(x-m)^T P^{-1}(x-m)\right) \tag{1.13}$$

The Kalman filter, derived using (1.5) and (1.6), can be expressed as the following recursive form:

$$p_{k-1}(x_{k-1}|z_{1:k-1}) = \mathcal{N}(x_{k-1}; x_{k-1|k-1}, P_{k-1}) \tag{1.14}$$
$$p_{k|k-1}(x_k|z_{1:k-1}) = \mathcal{N}(x_k; x_{k|k-1}, P_{k|k-1}) \tag{1.15}$$
$$p_{k|k}(x_k|z_{1:k}) = \mathcal{N}(x_k; x_{k|k}, P_{k|k}) \tag{1.16}$$

The Kalman filter is composed of two steps: the prediction step, where the state of the system is predicted given the previous measurements, and the update step, where the current state of the system is estimated with a new measurement at that time step. The corresponding means and covariances of the KF are given by [BLK01; RAG04; BWT11; Cha+11; GA14]:

1. Prediction:

$$x_{k|k-1} = F_{k-1}x_{k-1} \tag{1.17}$$
$$P_{k|k-1} = F_{k-1}P_{k-1}F_{k-1}^T + Q_{k-1} \tag{1.18}$$

2. Update:

$$S_k = HP_{k|k-1}H_k^T + R_k \tag{1.19}$$
$$K_k = P_{k|k-1}H_k^T S_k^{-1} \tag{1.20}$$
$$z_{k|k-1} = H_k x_{k|k-1} \tag{1.21}$$
$$x_{k|k} = x_{k|k-1} + K_k(z_k - z_{k|k-1}) \tag{1.22}$$
$$P_{k|k} = (I - K_k H_k)P_{k|k-1} \tag{1.23}$$

The matrix K_k is referred to as the Kalman gain. The term S_k is the covariance of the innovation term $\nu_k = z_k - z_{k|k-1}$. Under Gaussian assumptions on the dynamic and observation equations, an important property of the optimal Bayesian recursion is that a Gaussian posterior density at time $k-1$ will produce a Gaussian posterior density at time k. See Appendixes C and B for more details. Therefore, the *a posteriori* density at each time can be represented by a mean and a covariance matrix [BWT11; Cha+11].

1.1.3 The Extended Kalman filter

The Kalman filter (KF) is not applicable when the dynamic or measurements models are nonlinear as modeled by (1.1) and (1.3). In this case, the Extended Kalman filter (EKF) can be used, which consists of linearizing the model about the current mean and variance before applying the KF equations [Jaz70; GA14]. When the nonlinearities in the model are relatively weak, i.e., the nonlinear functions can be approximated by the first term in their Taylor series expansion, it is possible to approximate the predicted and posterior densities by Gaussian densities:

$$p_{k-1}(x_{k-1}|z_{1:k-1}) \approx \mathcal{N}(x_{k-1}; x_{k-1|k-1}, P_{k-1}) \tag{1.24}$$
$$p_{k|k-1}(x_k|z_{1:k-1}) \approx \mathcal{N}(x_k; x_{k|k-1}, P_{k|k-1}) \tag{1.25}$$
$$p_{k|k}(x_k|z_{1:k}) \approx \mathcal{N}(x_k; x_{k|k}, P_{k|k}) \tag{1.26}$$

Like the KF, the EKF has two steps: prediction and update. Thus, the corresponding means and covariances of the EKF are given by [Jaz70; BLK01; RAG04; BWT11; Cha+11; GA14]:

1. Prediction:

$$x_{k|k-1} = \mathrm{f}_{k-1}(x_{k-1}) \tag{1.27}$$
$$P_{k|k-1} = F_{k-1}P_{k-1}F_{k-1}^T + Q_{k-1} \tag{1.28}$$

2. Update:

$$S_k = HP_{k|k-1}H_k^T + R_k \tag{1.29}$$
$$K_k = P_{k|k-1}H_k^T S_k^{-1} \tag{1.30}$$
$$z_{k|k-1} = h_k(x_{k|k-1}) \tag{1.31}$$
$$x_{k|k} = x_{k|k-1} + K_k(z_k - z_{k|k-1}) \tag{1.32}$$
$$P_{k|k} = (I - K_kH_k)P_{k|k-1} \tag{1.33}$$

where the transition and the observation matrices are the local linearizations of the functions f_{k-1} and h_k, and are defined as the following Jacobians:

$$F_{k-1} = \left.\frac{\partial \mathrm{f}_{k-1}(x)}{\partial x}\right|_{x=x_{k-1|k-1}} \qquad H_k = \left.\frac{\partial h_k(x)}{\partial x}\right|_{x=x_{k|k-1}}$$

When the system and observation models have strong nonlinearities, the higher order terms of the Taylor expansion become significant and they cannot be safely ignored. In these cases the performance of the EKF becomes very poor [Jaz70; JUD95; GA14].

1.1.4 The Unscented Kalman filter

In the Extended Kalman filter (EKF), the state distribution is approximated by a Gaussian random variable, which is propagated analytically using the first-order linearization of the nonlinear system. This linearization can introduce large errors in the true posterior mean and covariance, which may lead to suboptimal performance and, in some cases, divergence of the filter. The Unscented Kalman filter (UKF) can be seen as an alternative to the EKF, which shares its computational complexity while avoiding the Jacobian computations [JUD95; JU97; JUD00; Wv00; JU04; van04; GA14].

The UKF addresses the approximation issues of the EKF by using a deterministic sampling approach called unscented transformation. The state distribution is approximated by a Gaussian using a minimal set of carefully chosen sample points. These sample points completely capture the true mean and covariance of the Gaussian random variable to the third-order (using Taylor series expansion) for any nonlinearity [JU97; Wv00; van04]. The unscented transformation is a method for calculating the statistics of a random variable that suffer a nonlinear transformation. Consider propagating the target state $x \in \mathbb{R}^n$ through a nonlinear function $z = h(x_k)$. Assume x has mean $\bar{x}$ and covariance P_{xx}. To calculate the statistics of z, form a matrix $\mathcal{X}$ of $2n+1$ sigma vectors $\mathcal{X}_i$ with corresponding weights $\mathcal{W}_i$, according to the following procedure [JU97; van04; GA14]:

Sample Values:

$$\begin{aligned} \mathcal{X}_0 &= \bar{x} && (1.34)\\ \mathcal{X}_i &= \bar{x} + \sqrt{(n+\lambda)P_{xx,i}} && 1 \le i \le n \\ \mathcal{X}_{n+i} &= \bar{x} - \sqrt{(n+\lambda)P_{xx,i}} && 1 \le i \le n \end{aligned}$$

Sample Weights:

$$\begin{aligned} \mathcal{W}_0^{(m)} &= \frac{\lambda}{(n+\lambda)} && (1.35)\\ \mathcal{W}_0^{(c)} &= \mathcal{W}_0^{(m)} + (1-\alpha^2+\beta) \\ \mathcal{W}_i^{(m)} &= \mathcal{W}_i^{(c)} = \frac{1}{2(n+\lambda)} && 1 \le i \le 2n \end{aligned}$$

where $\lambda = \alpha^2(n+\kappa) - n$ is a scaling parameter. The value of α, usually set to a small positive value (e.g., 10^{-3}), determines the spread of the sigma points around the mean $\bar{x}$. The term κ is a secondary scaling parameter that is usually set to 0, and β is used to incorporate prior knowledge of the distribution of x (the optimal value for Gaussian distributions is $\beta = 2$). These sigma vectors are propagated through the nonlinear function:

$$\mathcal{Z}_i = h(\mathcal{X}_i) \qquad 1 \le i \le 2n \tag{1.36}$$

and the mean $\bar{z}$ and the covariance P_{zz} of z are approximated using a weighted sample mean and covariance of the posterior sigma points:

$$\bar{z} \approx \sum_{i=0}^{2n} \mathcal{W}_i^{(m)} \mathcal{Z}_i \tag{1.37}$$

$$P_{zz} \approx \sum_{i=0}^{2n} \mathcal{W}_i^{(c)} (\mathcal{Z}_i - \bar{z})(\mathcal{Z}_i - \bar{z})^T \tag{1.38}$$

Note that no explicit calculation of Jacobians is necessary to implement the unscented transformation. Additionally, the overall number of computations are the same order as the EKF. This transformation results in approximations that are accurate to the third-order for Gaussian inputs. For non-Gaussian inputs, approximations are accurate to at least the second-order, and the accuracy of third and higher order moments is determined by the best choice of alpha and beta [JU97; van04]. The UKF is a straightforward extension of the unscented transformation to a recursive state estimation, where the state estimated is redefined as a concatenation of the original state and noise variables as follows [JU97; Wv00; GA14]:

$$x_k^a = \begin{bmatrix} x_k^T & v_{k-1}^T & w_k^T \end{bmatrix}^T$$

where v is the process noise vector with covariance matrix Q_k while w is the measurement noise vector with covariance matrix R_k. The unscented transformation and the prediction and update steps of the UKF can be summarized as below [JU97; Wv00; Jul02; GA14]:

Initialization:

$$x_k = \mathbb{E}[x_{k-1|k-1}]$$
$$P_k = \mathbb{E}[(x_{k-1|k-1} - x_k)(x_{k-1|k-1} - x_k)^T]$$

Initialization of the augumented state:

$$x_k^a = \mathbb{E}[x_{k-1|k-1}^a] = [x_k^T \quad 0 \quad 0]^T$$
$$P_k^a = \mathbb{E}[(x_{k-1|k-1}^a - x_k^a)(x_{k-1|k-1}^a - x_k^a)^T] = \begin{bmatrix} P_k & 0 & 0 \\ 0 & Q_k & 0 \\ 0 & 0 & R_k \end{bmatrix}$$

1. Calculate sigma-points:

$$\mathcal{X}_k^a = \begin{bmatrix} x_k^a & x_k^a + \sqrt{(n+\lambda)P_k^a} & x_k^a - \sqrt{(n+\lambda)P_k^a} \end{bmatrix} \tag{1.39}$$

2. Prediction:

$$\mathcal{X}_{k|k-1}^x = \mathrm{f}(\mathcal{X}_k^x, \mathcal{X}_k^v) \tag{1.40}$$
$$x_{k|k-1} = \sum_{i=0}^{2L} \mathcal{W}_i^{(m)} \mathcal{X}_{i,k|k-1}^x \tag{1.41}$$
$$P_{xx,k|k-1} = \sum_{i=0}^{2L} \mathcal{W}_i^{(c)} (\mathcal{X}_{i,k|k-1}^x - x_{k|k-1})(\mathcal{X}_{i,k|k-1}^x - x_{k|k-1})^T \tag{1.42}$$

3. Update:

$$\mathcal{Z}_{k|k-1} = h(\mathcal{X}_{k|k-1}^x, \mathcal{X}_k^w) \tag{1.43}$$
$$z_{k|k-1} = \sum_{i=0}^{2L} \mathcal{W}_i^{(m)} \mathcal{Z}_{i,k|k-1} \tag{1.44}$$
$$P_{zz,k|k} = \sum_{i=0}^{2L} \mathcal{W}_i^{(c)} (\mathcal{Z}_{i,k|k-1} - z_{k|k-1})(\mathcal{Z}_{i,k|k-1} - z_{k|k-1})^T \tag{1.45}$$
$$P_{xz,k|k} = \sum_{i=0}^{2L} \mathcal{W}_i^{(c)} (\mathcal{X}_{i,k|k-1}^x - x_{k|k-1})(\mathcal{Z}_{i,k|k-1} - z_{k|k-1})^T \tag{1.46}$$
$$K_k = P_{xz,k|k} P_{zz,k|k}^{-1} \tag{1.47}$$
$$x_{k|k} = x_{k|k-1} + K_k(z_k - z_{k|k-1}) \tag{1.48}$$
$$P_{xx,k|k} = P_{xx,k|k-1} - K_k P_{zz,k|k} K_k^T \tag{1.49}$$

where $\mathcal{X}^a = \begin{bmatrix} (\mathcal{X}^x)^T & (\mathcal{X}^v)^T & (\mathcal{X}^w)^T \end{bmatrix}$ are the augmented sigma-points, $k \in \{1, \cdots, \infty\}$, L is the dimension of the augmented states, f is the nonlinear dynamic function, h is the nonlinear observation function, and $\mathcal{W}_i$ are the weights calculated in (1.35).

1.1.5 The Particle filters

The advantage of the Kalman filter and its nonlinear extensions is that they have analytical forms, which makes them efficient to compute. Nevertheless, they are limited to the

use of Gaussian models to represent the process and measurement noise distributions. An alternative approach that does not need Gaussian assumptions is known as Particle filters.

Particle filters propagate representative samples of an estimated distribution, which can be viewed as "particles" that represent the probability densities [DGA00; DFG01; RAG04; GA14]. They are represented by a family of filters using different sample method that can improve performance on a given application. It has also been known as a Sequential Monte Carlo (SMC) [DGA00; DFG01] because the sampling may change as the solution progresses.

Suppose an arbitrary probability density $f_{k|k-1}(\cdot)$ is approximated by a set of random samples, $\left\{w_{k-1}^{(i)}, x_{k-1}^{(i)}\right\}_{i=1}^{N}$, where $x_{k-1}^{(i)}$ is the state of particle i with its respective weight $w_{k-1}^{(i)}$. The approximation of the probability density improves as $N \to \infty$, and the weights are normalized, i.e., $\sum_{i=1}^{N} w_{k-1}^{(i)} = 1$. The computation of the weights and particles is based on the idea of importance sampling. The initial particles at time k are drawn using the following equation [DFG01; RAG04; Aru+02]:

$$\tilde{x}_k^{(i)} \sim q_k(x_k | x_{k-1}^{(i)}, z_k) \quad 1 \leq i \leq N \tag{1.50}$$

where $q_k(\cdot)$ is called the importance density, whose weights are calculated in the following form:

$$w_k^{(i)} = \frac{\tilde{w}_k^{(i)}}{\sum_{j=1}^{N} \tilde{w}_k^{(j)}} \tag{1.51}$$

$$\tilde{w}_k^{(i)} = \tilde{w}_{k-1}^{(i)} \frac{g_k(z_k|\tilde{x}_k^{(i)}) f_{k|k-1}(\tilde{x}_k^{(i)}|\tilde{x}_{k-1}^{(i)})}{q_k(x_k|x_{k-1}^{(i)}, z_k)} \tag{1.52}$$

The method described above is known as sequential importance sampling, and in most cases fails after many iterations since almost all particle weights become zero due to particle degeneracy. This issue can be prevented by resampling the particles. The resampling step chooses N particles from $\left\{w_{k-1}^{(i)}, \tilde{x}_{k-1}^{(i)}\right\}_{i=1}^{N}$ based on their weights, i.e., the probability of the particle i being selected during resampling is equal to $w_k^{(i)}$. While this technique avoids degeneracy of particles, it leads to the loss of diversity because the particles with large weights are selected more often. To solve the problem of loss of diversity, a Markov Chain Monte Carlo (MCMC) step is recommended after resampling [RAG04; RBF16]. However, resampling is a computationally intensive operation and, to avoid repeating it on every iteration, adaptive resampling strategies can be used [MDJ12].

The definition of the importance density $q_k(\cdot)$ plays a crucial role in the implementation of the Particle filter. There are several designs of the algorithm in the literature, and the simplest way to define the importance density is to select $q_k(\cdot) \equiv f_{k|k-1}$, i.e., the importance density is equivalent to the Markov transition density as defined in (1.2). This Particle filter is referred as the Bootstrap filter [GSS93], and can lead to poor performance since most of the particles could be sampled from the region of the state space. A better approach is to use the information contained in the latest measurement z_k in the design of the importance density. Presented below is the simplest Bootstrap Particle filter to illustrate basic principles of the particle filtering [DGA00; DFG01; RAG04; RBF16]:

1. Initialization:

   i Define the number of particles N

   ii Draw initial samples:
   $$\tilde{x}_0^{(i)} \sim f_0(x_0) \qquad 1 \leq i \leq N \tag{1.53}$$

   iii Compute the initial weights:
   $$\tilde{w}_0^{(i)} = g_k(z_0|\tilde{x}_0^{(i)}) \tag{1.54}$$

   iv Normalize the importance weights:
   $$w_0^{(i)} = \frac{\tilde{w}_0^{(i)}}{\sum_{j=1}^{N} \tilde{w}_0^{(j)}} \tag{1.55}$$

2. For $k \in \{1, \cdots, \infty\}$:

   i Draw samples:
   $$\tilde{x}_k^{(i)} \sim q_k(x_k|x_{k-1}^{(i)}, z_k) \qquad 1 \leq i \leq N \tag{1.56}$$

   ii Update the weights:
   $$\tilde{w}_k^{(i)} = \tilde{w}_{k-1}^{(i)} \frac{g_k(z_k|\tilde{x}_k^{(i)}) f_{k|k-1}(\tilde{x}_k^{(i)}|\tilde{x}_{k-1}^{(i)})}{q_k(x_k|x_{k-1}^{(i)}, z_k)} \tag{1.57}$$

   iii Normalize the weights:
   $$w_k^{(i)} = \frac{\tilde{w}_k^{(i)}}{\sum_{j=1}^{N} \tilde{w}_k^{(j)}} \tag{1.58}$$

   iv Resampling, select index $j^i \in \{1, \cdots, N\}$ with probability $w_k^{(i)}$:
   $$x_k^{(i)} = \tilde{x}_k^{(j^i)} \tag{1.59}$$

1.2 Random Finite Set (RFS) for multi-object filtering

The standard Bayesian approach provides an optimal filtering solution for target state estimation when the number of targets is known and fixed, and when it is known which observation belongs to each target. When the number of targets to be estimated is unknown and time-varying, a probabilistic model of the underlying random process must be considered to accommodate variations in the number of targets as well as their states. A more natural model for this kind of system is called a point process, and it is the subject of a branch of probability theory known as stochastic geometry [SKM96; Ngu06].

For multi-object tracking applications, a more useful and realistic model is the simple-finite point process, also known as the Random Finite Set (RFS). The RFS theory was first systematically investigated in the 1970s by Kendall and Matheron, who carried out studies using statistical geometry [Ken74; Mat75]. In 1980, Serra applied the RFS theory to two-dimensional image analysis [Ser80] and, since then, this theory has become an essential tool in theoretical statistics, inspiring early practical works in data fusion. However, it was only in the mid-1990s with Mahler's work on Finite Set Statistics (FISST) and on the approximations

for the multi-object Bayesian filter that the RFS approach has been consistently adopted in the filtering community [Mah94a; Mah94b; GMN97].

The main idea behind the RFS approach is to treat the collection of individual states and observations as multi-object sets, and then to characterize uncertainties present in the multi-object tracking problem by modeling these sets as random finite sets as illustrated in Figure 1.3.

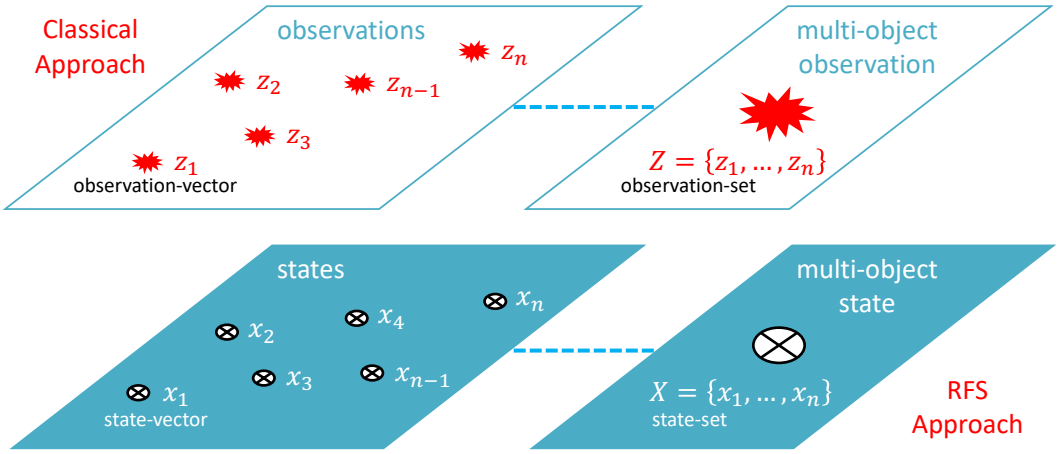

Figure 1.3: Comparison between the classical and the RFS representation of the multi-object state and observation. In the classical approach, random vectors are conveniently used to represent the state of the objects, however, this approach presents a missing representation of the uncertain number of objects. In the random finite set approach, an RFS naturally represents the uncertainty about the number of objects in a multi-object state and uses random vectors to represent the state of individual objects.

A random finite set X is defined as a finite set-valued random variable in which the number of elements (or cardinality) is random and the elements themselves are random, distinct, and unordered [DV88; SKM96; DV03; Ngu06; Mah07b; Mah14]. The cardinality of X is denoted by $|X|$, and is described by a discrete distribution $\rho(n) \in \mathbb{N}$, representing the probability that X has exactly n elements. For each cardinality $n \in \mathbb{N}$, there is a joint probability distribution $P_n(\cdot)$ describing the conditional density of the elements given that the cardinality of the set is equal to n. These joint distributions, $\rho(\cdot)$ and $P_n(\cdot)$, provide a complete specification of a random finite set [DV88; DV03; Ngu06].

For the single-object filtering, random vectors are conveniently used to represent the state of the objects. However, for the multi-object tracking, random vectors approach presents a missing representation of the uncertain number of objects. An RFS X defined as follows:

$$X = \{x^{(1)}, \cdots, x^{(n)}\} \tag{1.60}$$

consists of n unordered points with random object states $x^{(1)}, \cdots, x^{(n)}$ where $n \geq 0$ is a random number. An RFS naturally represents the uncertainty about the number of objects in a multi-object state and uses random vectors to represent the state of individual objects. Additionally, an RFS Z defined as follows:

$$Z = \{z^{(1)}, \cdots, z^{(m)}\} \tag{1.61}$$

is generally used to represent the measurement process which returns a random number of

measurements whose values $z^{(1)}, \cdots, z^{(m)}$ are also random. The randomness in the number of received measurements is due to the possibility of false alarms and missed detections. The RFS representation of the multi-object state and the measurement process are fundamental for the derivation of the multi-object Bayes filter (Section 1.2.6) which is an extension of the Bayes filter (Section 1.1.1) to multi-object tracking applications.

1.2.1 Random Finite Set notations and abbreviations

In order to distinguish random vectors from RFS, single-object states are denoted by small letters (e.g. x) while multi-object states are denoted by capital letters (e.g. X). Additionally, identical symbols are used for an RFS and its realization in the following. State spaces are represented by blackboard bold letters (e.g. $\mathbb{X}$). The finite subsets of a space X are denoted by $\mathcal{F}(X)$ and all possible subsets comprising exactly n elements are represented by $\mathcal{F}_n(X)$.

The inner product of two continuous functions $f(x)$ and $g(x)$ is abbreviated as follows:

$$\langle f, g \rangle \stackrel{\text{def}}{=} \int f(x)g(x)\mathrm{d}x \tag{1.62}$$

For discrete sequences, the above integral reduces to a sum. Additionally, the multi-object exponential notation is used to shorten the notation for real-valued functions $h(x)$, e.g. probability densities, which have to be evaluated for all state vectors of an RFS X. By definition, $h^X = 1$ in the case of $X = \emptyset$. The multi-object exponential notation is defined by [VV13]:

$$h^X \stackrel{\text{def}}{=} \prod_{x \in X} h(x) \tag{1.63}$$

The generalized Kronecker delta function is used to facilitate the application of the Kronecker delta function to arbitrary arguments like integers, vectors, and sets:

$$\delta_Y(X) \stackrel{\text{def}}{=} \begin{cases} 1 & \text{if } X = Y \\ 0 & \text{otherwise} \end{cases} \tag{1.64}$$

Additionally, the indicator or inclusion function is used to determine if a set X is a subset of Y:

$$1_Y(X) \stackrel{\text{def}}{=} \begin{cases} 1 & \text{if } X \subseteq Y \\ 0 & \text{otherwise} \end{cases} \tag{1.65}$$

1.2.2 Multi-object probability distributions

The spatial uncertainty of a random vector x is represented by a probability density function $p(x)$. Similarly, a multi-object probability density function $\pi(X)$ represents the incertitude about the multi-object state X which incorporates the uncertainty about the number of

objects and their states. The multi-object probability density depends on the number of objects represented by X and is given by [Mah07b; Mah14]:

$$\pi(X) = \begin{cases} \pi(\emptyset) & \text{if } X = \emptyset \\ \pi(\{x^{(1)}\}) & \text{if } X = \{x^{(1)}\} \\ \pi(\{x^{(1)}, x^{(2)}\}) & \text{if } X = \{x^{(1)}, x^{(2)}\} \\ \vdots & \vdots \end{cases} \tag{1.66}$$

Due to the dependency of $\pi(\cdot)$ on the number of objects, the evaluation of the integral over a multi-object probability density function requires the utilization of a set integral (see Appendix A):

$$\int \pi(X)\delta X = \sum_{i=0}^{\infty} \frac{1}{i!} \int_{\mathbb{X}^i} \pi(\{x^{(1)}, \cdots, x^{(i)}\}) \mathrm{d}x^{(1)} \cdots \mathrm{d}x^{(i)} \tag{1.67}$$

where $\pi(\{x^{(1)}, \cdots, x^{(i)}\}) = 0$ if $|\{x^{(1)}, \cdots, x^{(i)}\}| \neq i$. Since the evaluation of the set integral for all cardinalities is not computationally tractable, the computation of the set integral is evaluated only for a subset of all possible cardinalities.

The cardinality distribution of an RFS X provides an estimate for the number of objects represented by X, i.e., denotes the probability that X contains precisely n vectors, and is defined as follows:

$$\rho(n) = Pr(|X| = n) = \frac{1}{n!} \int \pi(\{x^{(1)}, \cdots, x^{(n)}\}) \mathrm{d}x^{(1)} \cdots \mathrm{d}x^{(n)} \tag{1.68}$$

There are many important classes of Random Finite Set (RFS) with their own unique properties. The most important classes of RFS found in multi-object filtering or tracking are summarized below:

Independent Identically Distributed Cluster RFS

The multi-object probability density of an independent and identically distributed (i.i.d.) cluster RFS with $X = \{x^{(1)}, \cdots, x^{(n)}\}$ is defined by:

$$\pi(X) = n!\rho(n)p(x^{(1)}) \cdots p(x^{(n)}) \tag{1.69}$$

where $\rho(n)$ is the cardinality probability distribution for $n \geq 0$, and $p(x^{(n)})$ is the probability density function representing the spatial distribution of the n-th object.

An i.i.d. cluster RFS X is an RFS that can be uniquely characterized by an intensity function $\nu(\cdot)$ and its cardinality distribution $\rho(n)$ [DV88; DV03]. The cardinality distribution of an i.i.d. cluster RFS can have an arbitrary form, with the only restriction being that its mean must match the integral of the intensity function $N = \sum_{n=0}^{\infty} n\rho(n) = \int \nu(x)\mathrm{d}x$. When

the individual elements of an RFS set are all distributed according to $\nu(\cdot)/N$, the multi-object probability density of an i.i.d. cluster can be rewritten as follows:

$$\pi(X) = n!\rho(n) \prod_{x \in X} \frac{\nu(x)}{N} \tag{1.70}$$

Poisson RFS

A Poisson RFS is a specific case of an i.i.d. cluster RFS, where the cardinality distribution $\rho(n)$ is restricted to the form of a Poisson distribution:

$$\rho(n) = e^{-\lambda} \frac{\lambda^n}{n!} \tag{1.71}$$

where λ represents the expected number of objects. In this way, the multi-object Poisson RFS is given by:

$$\pi(X) = e^{-\lambda} \lambda^n p(x^{(1)}) \cdots p(x^{(n)}) \tag{1.72}$$

An RFS with intensity $\nu(\cdot)$ is known as a Poisson RFS if its cardinality obeys a Poisson distribution $\rho(n)$, with mean $\lambda = \int \nu(x) \mathrm{d}x$ [MW03]. Consequently, a multi-object Poisson RFS is completely characterized by its intensity function $\nu(\cdot)$, since knowing the intensity makes possible to know the cardinality distribution. For a given cardinality, where the elements of an RFS set X are each independent and identically distributed with probability density $\nu(\cdot)/\lambda$, the multi-object Poisson RFS can be written as follows:

$$\pi(X) = e^{-\lambda} \prod_{x \in X} \nu(x) \tag{1.73}$$

The Poisson RFS is usually employed to represent processes exhibiting complete spatial randomness. In multi-object tracking applications, a Poisson RFS with intensity density:

$$\kappa(z) = \lambda_c \, c(z) \tag{1.74}$$

is commonly used to model the false alarm process with a mean number of λ_c false alarms. In general, the probability density $c(z)$ is assumed to be a uniform distribution over the complete measurement space due to the absence of precise information.

Bernoulli RFS

A Bernoulli RFS is used in cases where an RFS has probability $1 - r$ of being empty and probability r of holding a single element distributed according to a probability density $p(\cdot)$.

The Bernoulli RFS is completely characterized by the parameters r and p, and its density is defined accordingly:

$$\pi(X) = \begin{cases} 1-r & \text{if } X = \emptyset \\ rp(x) & \text{if } X = \{x\} \\ 0 & \text{if } |X| > 1 \end{cases} \tag{1.75}$$

The Bernoulli RFS is convenient in situations involving at most one target, where it is not certain whether the target is present.

Multi-Bernoulli RFS

The multi-Bernoulli RFS is the union of a fixed number of independent Bernoulli RFS, where the elements are statistically independent of each other, and the i-th element has probability of existence $r^{(i)}$ and spatial probability density $p^{(i)}$, for $i \in [1, \cdots, M]$. The multi-Bernoulli of an RFS X is completely defined by the set of parameters $\left\{(r^{(i)}, p^{(i)})\right\}_{i=1}^{M}$, and its density can be described by [Mah07b; Mah14]:

$$\pi(\{x^{(1)}, \cdots, x^{(n)}\}) = \begin{cases} \prod_{j=1}^{M} \left(1 - r^{(j)}\right) & n = 0 \\ \prod_{j=1}^{M} \left(1 - r^{(j)}\right) \sum_{1 \le i_1 \ne \cdots \ne i_n \le M} \prod_{j=1}^{n} \frac{r^{(i_j)} p^{(i_j)}(x^{(j)})}{1 - r^{(i_j)}} & 1 \le n \le M \\ 0 & n > M \end{cases} \tag{1.76}$$

where the probability density function for $n = 0$ and $X = \emptyset$, i.e. $\pi(\emptyset) = \prod_{j=1}^{M} (1 - r^{(j)})$, corresponds to the probability that none of the M objects exists. In order to obtain the probability density function for $X = \{x^{(1)}, \cdots, x^{(n)}\}$, it is necessary to sum over all n possible permutations of the vector $x^{(i)}$.

The multi-Bernoulli RFS is convenient in cases involving multi-objects where the presence of any particular object is not assured. The cardinality distribution of a multi-Bernoulli RFS X is obtained by ignoring the spatial distribution in (1.76):

$$\rho(n) = \prod_{j=1}^{M} \left(1 - r^{(j)}\right) \sum_{1 \le i_1 \ne \cdots \ne i_n \le M} \prod_{j=1}^{n} \frac{r^{(i_j)}}{1 - r^{(i_j)}} \tag{1.77}$$

According to Mahler, the Probability Hypothesis Density (PHD) of a multi-Bernoulli RFS is given by [Mah07b; Mah14]:

$$\nu(x) = \sum_{j=1}^{M} r^{(j)} p^{(j)}(x) \tag{1.78}$$

and consequently, the mean cardinality of a multi-Bernoulli can be calculate by:

$$\hat{N} = \sum_{j=1}^{M} r^{(j)} \tag{1.79}$$

1.2.3 Multi-object filtering using RFS

In the multi-object filtering problem, not only do the states of the objects are time-varying, but the number of items also changes due to objects that can appear and disappear. The targets can be observed by different types of sensor (or sensors) such as radar, sonar, lidar, camera, and infrared camera, and the sensor signals are preprocessed into a set of points or detections. Thus, the goal is to jointly estimate the number of objects and their states from the accumulated observations. In the random finite set framework, the objective is to estimate an RFS of multi-object states $X_k \in \mathcal{X}$ at time k: in other words, to estimate the cardinality $|X_k| = n$ and the constituent object states $x_{i,k}, i \in [1, \cdots, n]$ given a sequence of measurement sets $Z_{1:k} = \{Z_1, \cdots, Z_k\}$. The symbols $\mathcal{X}$ and $\mathcal{Z}$ represent the state and observation spaces. Figure 1.4 shows a schematic representation of the multi-object filtering using random finite sets.

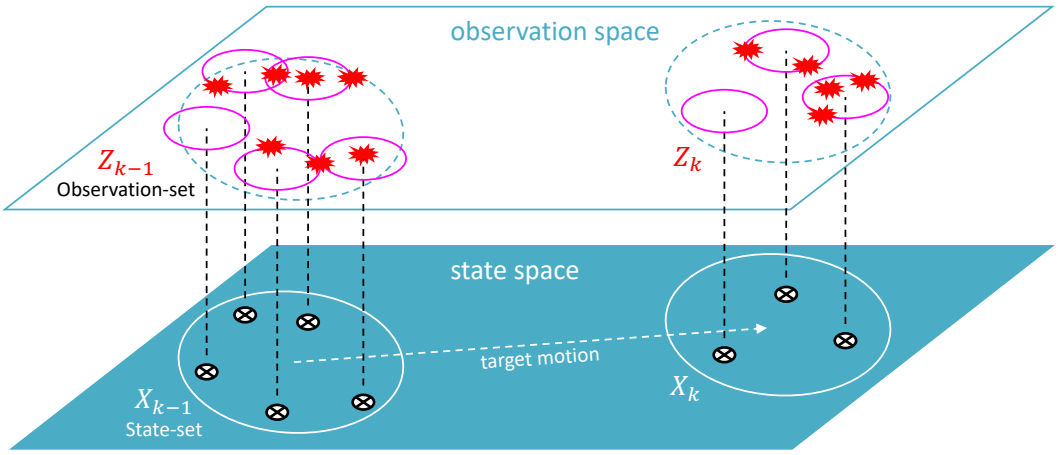

Figure 1.4: Simplified schematic of a dynamic model where a multi-object state evolves in the state space. In this example, the number of objects in the state space changes from five at time $k-1$ to three at time k, and each target may generate a random number of measurements due to clutter and false alarms, while some targets may not be detected. As a result, at each time k, the multi-object observation is a set of detections where only some are generated by the targets, and there is no information on which targets generated which detections.

Modeling states and observations as random finite sets is essential to generalizing the single-object Bayes filter to the multi-object Bayes filter, because the RFS framework naturally models the time evolution of the multi-object state as well as the object dynamic, birth, death, and spawning. Thus, random finite sets are used to model uncertainty in the state $X_k \subset \mathcal{X}$ and observation $Z_k \subset \mathcal{Z}$ at each time k. Due to imperfections of sensors, it is possible that at time k, some of the targets in X_k are not detected. Furthermore, the observation set Z_k normally includes false alarms or clutter in addition to target detection. Considering that the number of objects and observations are variables, it is mathematically reasonable to represent the collections of states and observations as random finite sets as shown below [Mah03a; Mah07b; Mah14]:

$$X_k = \{x_{1,k}, \cdots, x_{M(k),k}\} \subset \mathcal{X}$$
$$Z_k = \{z_{1,k}, \cdots, z_{N(k),k}\} \subset \mathcal{Z}$$

where $M(k)$ is the number of existing targets at time k, while $N(k)$ is the number of current measurements gathered by a sensor at time k. Random finite sets for target and observation spaces are represented by $\mathcal{F}(\mathcal{X})$ and $\mathcal{F}(\mathcal{Z})$, where $\mathcal{F}(\cdot)$ denotes the set of all finite subsets.

1.2.4 Multi-object dynamic model

In general, multi-object filtering algorithms assume a standard multi-object transition model, where every existing target state $x_{k-1} \in X_{k-1}$, either continues to exist at time k with probability $P_{S,k|k-1}(x_{k-1})$ or dies with probability $1 - P_{S,k|k-1}(x_{k-1})$. Each surviving target moves to a new state x_k with probability density $f_{k|k-1}(x_k|x_{k-1})$, while each target that dies becomes an empty set $\emptyset$. Thus, the set of surviving objects, represented by the RFS $S_{k|k-1}(X_{k-1})$, is modeled by a multi-Bernoulli RFS as defined in (1.76) with parameter set $\left\{\left(P_{S,k|k-1}(x_{k-1}), f_{k|k-1}(x_k|x_{k-1})\right)\right\}$.

Additionally, a random number of objects can appear or disappear from random locations in the state space at time k. New objects may appear either by spontaneous birth independently of existing objects or by spawning from an existing object. Consequently, the set of born objects at time k is represented by the RFS Γ_k, whose model depends on the particular filtering algorithm in use. Different multi-object filtering approaches employ different models for target births and deaths. The set of spawned objects at time k are denoted by $B_{k|k-1}(X_{k-1})$. It is assumed that the RFS $S_{k|k-1}(\cdot)$, $B_{k|k-1}(\cdot)$ and Γ_k are independents. Consequently, the multi-object state modeled by the RFS X_k and illustrated in Figure 1.5, is defined accordingly:

$$X_k = S_{k|k-1}(X_{k-1}) \cup B_{k|k-1}(X_{k-1}) \cup \Gamma_k \tag{1.80}$$

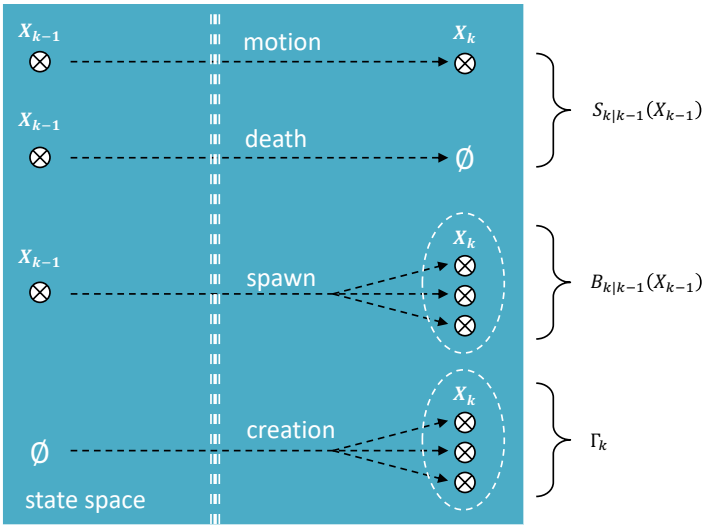

Figure 1.5: Multi-object dynamical model assumptions.

Let $\pi_{S,k|k-1}(\cdot)$ and $\pi_{B,k|k-1}(\cdot)$ denote the density of $S_{k|k-1}$ and $B_{k|k-1}$ respectively, then, the predicted density of X_k can be represented by the following convolution:

$$f_{k|k-1}(X_k|X_{k-1}) = \sum_{W \subseteq X_k} \pi_{S,k|k-1}(W|X_{k-1})\pi_{B,k|k-1}(X_k - W) \tag{1.81}$$

1.2.5 Multi-object observation model

The observation model incorporates the measurement likelihood, object detection uncertainty at the sensor, and false alarms or clutter. Each object $x_k \in X_k$ either is detected with probability $P_{D,k|k}(x_k)$ and generates an observation z_k with likelihood $g_k(z_k|x_k)$, or is missed with probability $1 - P_{D,k|k}(x_k)$ and generates an empty set $\emptyset$. It is assumed that the object measurements are conditionally independent; thus, the RFS of detections, represented by $D_k(X_k)$, is modeled by a multi-Bernoulli RFS with parameter set $\left\{\left(P_{D,k|k}(x_k), g_k(z_k|x_k)\right)\right\}$.

In addition to the detections of real objects, the sensor can gather a random number of false alarms from random locations in the measurement space. The set of false alarms at time k is represented by an RFS K_k, which, in general, is modeled as a Poisson RFS. The multi-object observation modeled by the RFS Z_k and illustrated in Figure 1.6 is defined by:

$$Z_k = D_k(X_k) \cup K_k \tag{1.82}$$

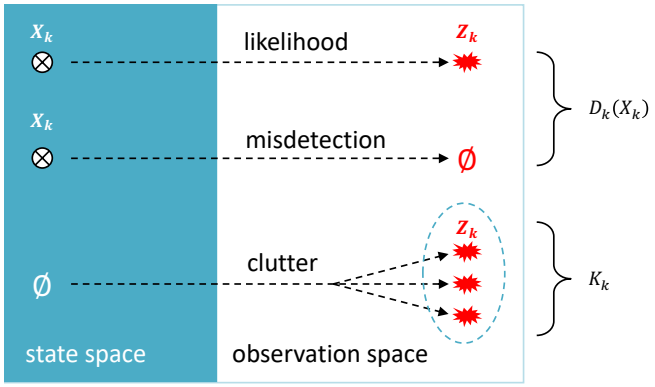

Figure 1.6: Multi-object observation model assumptions.

Let $\pi_{D,k}(\cdot)$ and $\pi_{K,k}(\cdot)$ denote the density of D_k and K_k respectively, then, the density of Z_k can be defined by the following convolution:

$$g_k(Z_k|X_k) = \sum_{W \subseteq Z_k} \pi_{D,k}(W|X_k)\pi_{K,k}(Z - W) \tag{1.83}$$

1.2.6 The Multi-object Bayes filter

Once the multi-object state and observation are modeled as an RFS, the estimate can be recursively calculated using the multi-object Bayesian filtering. The idea is to jointly estimate the number of objects and their states from the noisy observations. Thus, the objective of

the multi-object Bayesian estimator is to determine the posterior density of multi-object state $\pi_k(X_k|Z_{1:k})$ at each time k, where $Z_{1:k} = \{Z_1, \cdots, Z_k\}$ denotes the observation set sequence up to time k. The multi-object posterior density can be computed recursively via the prediction and update steps. The predicted density at time k, denoted as $\pi_{k|k-1}(X_k|Z_{1:k-1})$, is computed by the multi-object Chapman-Kolmogorov equation [GMN97; Mah03a; Mah07b; Mah14]:

$$\pi_{k|k-1}(X_k|Z_{1:k-1}) = \int f_{k|k-1}(X_k|X_{k-1})\pi_{k-1}(X_{k-1}|Z_{1:k-1})\delta X_{k-1} \tag{1.84}$$

When a new observation set Z_k is received at time k, the new posterior density is computed via the multi-object Bayes' rule [Shi84; DV88; DV03; Ros14]:

$$\pi_{k|k}(X_k|Z_{1:k}) = \frac{g_k(Z_k|X_k)\pi_{k|k-1}(X_k|Z_{1:k-1})}{\int g_k(Z_k|X_k)\pi_{k|k-1}(X_k|Z_{1:k-1})\delta X_k} \tag{1.85}$$

The multi-object Bayes filter requires two *a priori* distributions: the multi-object Markov transition density $f_{k|k-1}(X_k|X_{k-1})$, which is the probability density that the multi-object state will move from state X_{k-1} at time $k-1$ to state X_k at time k, and the multi-object likelihood function $g_k(Z_k|X_k)$, which is the probability density that the observation set Z_k will be collected if the objects at time k have state set X_k. Given a sequence of observation sets, the multi-object Bayes filter propagates the probability density of the multi-object RFS over time as illustrated by the following diagram:

$$\cdots \rightarrow \pi_{k-1}(X_{k-1}|Z_{1:k-1}) \xrightarrow{\text{prediction}} \pi_{k|k-1}(X_k|Z_{1:k-1}) \xrightarrow{\text{update}} \pi_{k|k}(X_k|Z_{1:k}) \rightarrow \cdots$$

The multi-object Bayes equations (1.84) and (1.85) look very similar to the single-object Bayes equations (1.5) and (1.6), and the difference is that the multi-object Bayes equations use set-integrals and the functions involved have units whereas the former are unitless (for more details, see Appendix A). Thus, due to the combinatorial nature of multi-object densities and set-integrals, the multi-object Bayes recursion is intractable in most practical applications. However, it serves as the basis for deriving multi-object filtering and tracking algorithms using the RFS framework.

1.2.7 The Probability Hypothesis Density (PHD) filter

In a single-object filtering problem, two essential statistics of the posterior density are the first-order moment, a vector representing the posterior expectation, and the second-order moment, a matrix representing the covariance. In general, these two order moments are assumed to be sufficient statistics for the posterior density, and higher-order moments are ignored. In some specific cases, even the second-order moment can be neglected, and the posterior density can be described by only the first-order moment, similarly to the Kalman filter approach. However, representing a complete density by its first-order moment is plausible if the density is unimodal and has low variance.

Representing and propagating a density with only the first-order moment is useful, and the same logic has been extended to derive multi-object filters within the random finite set

formalism. In 2000, Mahler proposed approximating the multi-object Bayes recursion by propagating a first-order statistical moment, the posterior expectation, in the place of the posterior distribution [Mah00; Mah01a; Mah01b]. For an RFS X on $\mathcal{X}$, with a probability distribution π, its first-order moment is a non-negative function ν on $\mathcal{X}$, called intensity or Probability Hypothesis Density (PHD) function, defined by the following expectation:

$$\nu(x) = \mathbb{E}[\delta_X(x)] = \int \delta_X(x)\pi(X)\delta X \tag{1.86}$$

where the Dirac delta function $\delta_X(x)$ is defined as:

$$\delta_X(x) = \begin{cases} 0 & \text{if } X = \emptyset \\ \sum_{w \in X} \delta_X(w) & \text{otherwise} \end{cases} \tag{1.87}$$

The PHD function is defined based on the single-object state space, and has the primary property that its integral for any closed subsets $S \subseteq \mathcal{X}$ gives the expected number of objects contained in S. The PHD is not a probability density, since its integral may result in a value greater than one:

$$\int_S \nu(x)\mathrm{d}x = \mathbb{E}[|X \cap S|] \tag{1.88}$$

Using the first-order moment of an RFS (in other words, the PHD function), it is possible to develop a computationally tractable approximation to the multi-object Bayes filter [Mah03a]. This approximation, called the PHD filter, is a recursive process that propagates the intensity, or the first-order moment, of the target random finite set. A Poisson assumption is made to derive a closed-form solution for the PHD filter. The predictive density is approximated by a Poisson point process to estimate potentially many targets, including the birth, death, and spawning of targets automatically. The PHD filter approximates the predicted and updated probability densities $\pi_{k|k-1}(X_k|Z_{1:k-1})$ and $\pi_{k|k}(X_k|Z_{1:k})$ with the respective first-order moment densities, denoted by $\nu_{k|k-1}(X_k|Z_{1:k-1})$ and $\nu_{k|k}(X_k|Z_{1:k})$, which are propagated over time as illustrated by the following diagram:

$$\begin{array}{ccccccc} \cdots \rightarrow & \pi_{k-1}(X_{k-1}|Z_{1:k-1}) & \xrightarrow{\text{prediction}} & \pi_{k|k-1}(X_k|Z_{1:k-1}) & \xrightarrow{\text{update}} & \pi_{k|k}(X_k|Z_{1:k}) \rightarrow \cdots \\ & \downarrow & & \downarrow & & \downarrow \\ \cdots \rightarrow & \nu_{k-1}(x_{k-1}|Z_{1:k-1}) & \xrightarrow{\text{prediction}} & \nu_{k|k-1}(x_k|Z_{1:k-1}) & \xrightarrow{\text{update}} & \nu_{k|k}(x_k|Z_{1:k}) \rightarrow \cdots \end{array}$$

Suppose here that ν_{k-1} is the posterior PHD at time $k-1$, $\gamma_k(\cdot)$ is the intensity of spontaneous target births at time k, $P_{S,k}(x_{k-1})$ is the probability that a target still exists at time k, $f_{k|k-1}$ is the single-object Markov transition density from time $k-1$ to time k, and $\beta_{k|k-1}(x|x_{k-1})$ is the intensity of the object spawned from another object with previous state x_{k-1} at time k. The probability that an object with state x_k is detected is $P_{D,k}(x_k)$, and a set of observations Z_k is received at time k, which contains false alarms with intensity κ_k. The posterior intensity can then be propagated via the PHD recursion [Mah03a; Mah07b;

Mah14]:

$$\nu_{k|k-1}(x) = \gamma_k(x_k) + \int \Big(P_{S,k}(x_{k-1}) f_{k|k-1}(x_k|x_{k-1}) + \beta_{k|k-1}(x|x_{k-1})\Big)\nu_{k-1}(x_{k-1})\mathrm{d}x_{k-1} \tag{1.89}$$

$$\nu_{k|k}(x_k) = \Big(1 - P_{D,k}(x_k)\Big)\nu_{k|k-1}(x_k) + \sum_{z\in Z_k} \frac{P_{D,k}(x_k) g_k(z|x_k)\nu_{k|k-1}(x_k)}{\kappa_k(z) + \int P_{D,k}(x_k) g_k(z|x_k)\nu_{k|k-1}(x_k)\mathrm{d}x_k} \tag{1.90}$$

There are many implementations of the PHD filter. The first is based on the sequential Monte Carlo method, and was independently proposed in 2003 by Sidenbladh [Sid03], Zajic and Mahler [Tim03], and Vo, Singh, and Doucet [VSD03]. This is called the Sequential Monte Carlo PHD (SMC-PHD) filter, and its analysis of convergence was realized by Clark and Johansen [CB06; Joh+06]. In 2006, Vo and Ma proposed the first analytic implementation of the PHD filter based on Gaussian mixtures, the Gaussian-Mixture PHD (GM-PHD) filter, which included an application for linear-Gaussian models based on Kalman filters, and for nonlinear-Gaussian models based on the EKF or UKF [VM05; VM06]. Convergence analysis of the GM-PHD filter, including a version based on the Gaussian Particle filter, was presented by Clark [CVV07]. In 2010, Macagnano proposed a PHD version based on the cubature Kalman filter to deal with nonlinearity existing between observations and state model [Md10].

1.2.8 The Gaussian-mixture PHD filter

The Probability Hypothesis Density (PHD) filter given in (1.89)-(1.90) does not, in general, admit closed-form solutions. Under linear Gaussian assumptions, the multi-object posterior density can be approximated using the Gaussian-Mixture PHD (GM-PHD) filter [VM05; VM06]. The linear Gaussian multi-object model includes certain assumptions about the birth, death and detection of targets [VM06].

Each target follows a linear Gaussian dynamic model, i.e.:

$$f_{k|k-1}(x_k|x_{k-1}) = \mathcal{N}(x_k; F_{k-1}x_{k-1}, Q_{k-1}) \tag{1.91}$$
$$g_k(z_k|x_k) = \mathcal{N}(z_k; H_k x_k, R_k) \tag{1.92}$$

where the term $\mathcal{N}(\cdot; m, P)$ is the Gaussian density with mean m and covariance P, F_{k-1} is the state transition matrix, Q_{k-1} is the process noise covariance, H_k is the observation matrix, and R_k is the observation noise covariance.

The detection and survival probabilities are both state-independent:

$$p_{D,k}(x) = p_{D,k} \tag{1.93}$$
$$p_{S,k}(x) = p_{S,k} \tag{1.94}$$

The intensities of the birth and spawn RFSs are both Gaussian mixtures of the form:

$$\gamma_k(x_k) = \sum_{i=1}^{J_{\gamma,k}} w_{\gamma,k}^{(i)} \mathcal{N}\left(x_k; m_{\gamma,k}^{(i)}, P_{\gamma,k}^{(i)}\right) \tag{1.95}$$

$$\beta_{k|k-1}(x_k|x_{k-1}) = \sum_{j=1}^{J_{\beta,k}} w_{\beta,k}^{(j)} \mathcal{N}\left(x_k; F_{\beta,k-1}^{(j)} x_{k-1} + d_{\beta,k-1}^{(j)}, Q_{\beta,k-1}^{(j)}\right) \tag{1.96}$$

where $w_{\gamma,k}^{(i)}$, $m_{\gamma,k}^{(i)}$, $P_{\gamma,k}^{(i)}$, and $J_{\gamma,k}$ are, respectively, the weights, means, covariance, and total number of Gaussian mixtures for birth intensity, and $w_{\beta,k}^{(j)}$, $F_{\beta,k-1}^{(j)} + d_{\beta,k-1}^{(j)}$, $Q_{\beta,k-1}^{(j)}$, and $J_{\beta,k}$ are, respectively, the weights, means, covariance, and total number of Gaussian mixtures for spawning intensity.

The prediction and update steps of the GM-PHD filter can be described as follows [VM05; VM06]:

1. Prediction:

   Given $\{w_{k-1}^{(i)}, m_{k-1}^{(i)}, P_{k-1}^{(i)}\}$ and the measurement Z_k, suppose that the posterior intensity at time $k-1$ is a Gaussian mixture of the form:

   $$\nu_{k-1}(x) = \sum_{i=1}^{J_{k-1}} w_{k-1}^{(i)} \mathcal{N}\left(x; m_{k-1}^{(i)}, P_{k-1}^{(i)}\right) \tag{1.97}$$

   Then, considering $\gamma_k(x)$ defined in (1.95), the predicted intensity for time k is also a Gaussian mixture given by:

   $$\nu_{k|k-1}(x) = \nu_{S,k|k-1}(x) + \nu_{\beta,k|k-1}(x) + \gamma_k(x) \tag{1.98}$$

   where:

   $$\nu_{S,k|k-1}(x) = p_{S,k} \sum_{j=1}^{J_{k-1}} w_{k-1}^{(j)} \mathcal{N}\left(x; m_{S,k|k-1}^{(j)}, P_{S,k|k-1}^{(j)}\right) \tag{1.99}$$

   $$m_{S,k|k-1}^{(j)} = F_{k-1} m_{k-1}^{(j)} \tag{1.100}$$

   $$P_{S,k|k-1}^{(j)} = Q_{k-1} + F_{k-1} P_{k-1}^{(j)} F_{k-1}^T \tag{1.101}$$

   $$\nu_{\beta,k|k-1}(x) = \sum_{j=1}^{J_{k-1}} \sum_{\ell=1}^{J_{\beta,k}} w_{k-1}^{(j)} w_{\beta,k-1}^{(\ell)} \mathcal{N}\left(x; m_{\beta,k|k-1}^{(j,\ell)}, P_{\beta,k|k-1}^{(j,\ell)}\right) \tag{1.102}$$

   $$m_{\beta,k|k-1}^{(j,\ell)} = F_{\beta,k-1}^{(\ell)} m_{k-1}^{(j)} + d_{\beta,k-1}^{(\ell)} \tag{1.103}$$

   $$P_{\beta,k|k-1}^{(j,\ell)} = Q_{\beta,k-1}^{(\ell)} + F_{\beta,k-1}^{(\ell)} P_{\beta,k-1}^{(j)} (F_{\beta,k-1}^{(\ell)})^T \tag{1.104}$$

2. Update:

   $$\nu_{k|k}(x) = (1 - p_{D,k}) \nu_{k|k-1}(x) + \sum_{z \in Z_k} \nu_{D,k}(x; z) \tag{1.105}$$

   where:

$$\nu_{D,k}(x;z) = \sum_{j=1}^{J_{k|k-1}} w_k^{(j)}(z)\,\mathcal{N}\left(x; m_{k|k}^{(j)}(z), P_{k|k}^{(j)}\right) \tag{1.106}$$

$$w_k^{(j)}(z) = \frac{p_{D,k} w_{k|k-1}^{(j)} q_k^{(j)}(z)}{\kappa_k(z) + p_{D,k}\sum_{\ell=1}^{J_{k|k-1}} w_{k|k-1}^{(\ell)} q_k^{(\ell)}(z)} \tag{1.107}$$

$$q_k^{(j)}(z) = \mathcal{N}\left(z; H_k m_{k|k-1}^{(j)}, R_k + H_k P_{k|k-1}^{(j)} H_k^T\right) \tag{1.108}$$

$$m_{k|k}^{(j)}(z) = m_{k|k-1}^{(j)} + K_k^{(j)}\left(z - H_k m_{k|k-1}^{(j)}\right) \tag{1.109}$$

$$P_{k|k}^{(j)} = \left[I - K_k^{(j)} H_k\right] P_{k|k-1}^{(j)} \tag{1.110}$$

$$K_k^{(j)} = P_{k|k-1}^{(j)} H_k^T (H_k P_{k|k-1}^{(j)} H_k^T + R_k)^{-1} \tag{1.111}$$

1.2.9 Optimal Subpattern Assignment (OSPA) metric

Since there are several implementations of the PHD filter and different algorithms based on RFS framework, a metric to analyze and compare the performance of these algorithms becomes necessary. Also to the state estimation error considered by single-object miss distances like the Euclidean or Mahalanobis distance, the metric should incorporate the cardinality error.

The Optimal Mass Transfer (OMAT) metric was one of the first approaches for a multi-object miss distance [HM04]. The OMAT is based on the Wasserstein distance and resolves some issues of the Hausdorff metric in multi-object filtering applications. This metric is not defined if one of the two sets is empty and it does not penalize multiple estimates for a single object. To overcome the OMAT weaknesses, Schuhmacher et al. proposed a new metric, the Optimal SubPattern Assignment (OSPA) metric [SVV08].

The OSPA is a metric used to quantify the distance between two finite sets, allowing to compare results between different algorithms based on the RFS framework [SVV08]. This metric accounts for differences in the cardinality and the individual elements between any two finite sets X and Y in a mathematically consistent way. For any two state vectors x and y the distance is calculated according:

$$d_c(x,y) = \min(c, d(x,y)) \tag{1.112}$$

where $c > 0$ is the cut-off parameter. For two finite sets $X = \{x^{(1)}, \cdots, x^{(m)}\}$ and $Y = \{y^{(1)}, \cdots, y^{(n)}\}$, where $m \leq n$, the OSPA distance of order $p \geq 1$ and cut-off $c > 0$ is given by:

$$d_p^{(c)}(X,Y) = \begin{cases} 0 & m = n = 0 \\ \left(\frac{1}{n}\left(\min\limits_{\pi\in\Pi_n}\sum\limits_{i=1}^{m} d_c(x_i, y_{\pi(i)})^p + c^p(n-m)\right)\right)^{\frac{1}{p}} & m \leq n \\ d_p^{(c)}(Y,X) & m > n \end{cases} \tag{1.113}$$

where Π_n is the set of permutations on $\{1, \cdots, n\}$. In the case of $m > n$, the parameters X and Y have to be switched. The order parameter p determines the sensitivity of the metric in penalizing outlier estimates, and the cut-off parameter c restricts the maximum distance.

In the context of multi-object performance evaluation, it is possible to calculate the localization error using the OSPA metric as follows:

$$d_p^{(c,\ \mathrm{loc})}(X, Y) = \left(\frac{1}{n} \min_{\pi \in \Pi_n} \sum_{i=1}^{m} d^c(x_i, y_{\pi(i)})^p \right)^{\frac{1}{p}} \tag{1.114}$$

and the cardinality error:

$$d_p^{(c,\ \mathrm{card})}(X, Y) = \left(\frac{c^p(n-m)}{n} \right)^{\frac{1}{p}} \tag{1.115}$$

Although the separated errors are no longer a metric on the space of finite subsets, and they provide only additional information for further improvements of the multi-object tracking algorithms [SVV08].

1.3 Random Finite Set (RFS) for multi-object tracking

The Probability Hypothesis Density (PHD) filter was proposed by Mahler for jointly estimating the time-varying number of targets and their states from a noisy sequence of measurements [Mah03a]. Under linear Gaussian assumptions, the multi-object posterior density can be approximated using the Gaussian-Mixture PHD (GM-PHD) filter [VM05; VM06]. The original implementation of the GM-PHD filter provided estimates for the set of objects states but did not ensure continuity of the individual object tracks [CPV06; PVC06].

In the GM-PHD filter, the posterior intensity function is approximated by a sum of weighted Gaussians whose means, weights, and covariances can be propagated analytically in time. In general, the means and covariances are estimated using a Kalman filter. The algorithm to determine the targets uses the weights of the Gaussians and does not take into account temporal continuity. However, the trajectories of the targets can be determined by taking into account the evolution of the Gaussian mixture.

To accurately estimate the number of objects and their states at each time, it is also essential to know the trajectory of each object and distinguish it from different objects. There are several methods to create the objects' trajectory reported in the literature. Panta et al. used the PHD filter to pre-filter the data input to a Multiple Hypothesis Tracking (MHT) algorithm [Pan+04]. Clark suggested associating target estimates between iterations, also known as estimate-to-track association [CB05]. For the Sequential Monte Carlo PHD (SMC-PHD) filter, Panta et al. proposed partitioning particle data to assign labels to particles within the same cluster as well as linking clusters between time frames if there is a large intersection of particles with the same label from the previous time [PVS05]. This latter technique uses the empirical PHD distribution and is similar to the technique used in the development of a tracker algorithm based on the GM-PHD, the Gaussian-Mixture PHD Tracker (GM-PHDT).

1.3.1 The Probability Hypothesis Density Tracker (PHDT)

The Gaussian-Mixture PHD Tracker (GM-PHDT) is an extension of the GM-PHD filter. The GM-PHDT has the ability to estimate the number of targets, track the trajectories of the targets over time, operate with missed detections, and give the trajectories of the targets in the past once a target has been identified.

In the GM-PHDT each target follows a linear Gaussian dynamical model as described in (1.91) and (1.92). The GM-PHDT algorithm is initialized and then iterates through prediction, update, pruning, merging, and target state estimation steps:

1. Initialization:

   At time $k = 0$, the initial intensity ν_0 is the sum of J_0 Gaussians,

   $$\nu_0(x) = \sum_{i=1}^{J_0} w_0^{(i)} \mathcal{N}\left(x; m_0^{(i)}, P_0^{(i)}\right) \tag{1.116}$$

   These are distributed across the state space where each Gaussian $\mathcal{N}(x; m_0^{(i)}, P_0^{(i)})$ has mean state vector $m_0^{(i)}$, covariance $P_0^{(i)}$ and weight $w_0^{(i)}$. An identifier or tag is assigned to each Gaussian to form the following set:

   $$T_0 = \left\{\tau_0^{(1)}, \cdots, \tau_0^{(J_0)}\right\} \tag{1.117}$$

   where $\tau_k^{(j)}$ denotes the tag of the j^{th} Gaussian.

2. Prediction:

   The predicted intensity to time k is a Gaussian mixture of the form:

   $$\nu_{k|k-1}(x) = \nu_{S,k|k-1}(x) + \gamma_k(x) \tag{1.118}$$

   where:

   $$\nu_{S,k|k-1}(x) = p_{S,k} \sum_{j=1}^{J_{k-1}} w_{k-1}^{(j)} \mathcal{N}\left(x; m_{S,k|k-1}^{(j)}, P_{S,k|k-1}^{(j)}\right) \tag{1.119}$$

   $$m_{S,k|k-1}^{(j)} = F_{k-1} m_{k-1}^{(j)} \tag{1.120}$$

   $$P_{S,k|k-1}^{(j)} = Q_{k-1} + F_{k-1} P_{k-1}^{(j)} F_{k-1}^T \tag{1.121}$$

   The term $\beta_{k|k-1}$ for spawned objects has been omitted to keep the explanation simple.

   In this step it is necessary to concatenate the set of tags from the previous time with new tags from the Gaussians introduced by the spontaneous birth model:

   $$T_{k|k-1} = T_k \cup \left\{\tau_{\gamma k}^{(1)}, \cdots, \tau_{\gamma k}^{(J_{\gamma k})}\right\} \tag{1.122}$$

   where $\tau_{\gamma k}^{(j)}$ is the tag assigned to the j^{th} Gaussian generated by γ_k.

3. Update:

$$\nu_{k|k}(x) = (1 - p_{D,k})\nu_{k|k-1}(x) + \sum_{z \in Z_k} \nu_{D,k}(x; z) \tag{1.123}$$

where:

$$\nu_{D,k}(x; z) = \sum_{j=1}^{J_{k|k-1}} w_k^{(j)}(z)\mathcal{N}\left(x; m_{k|k}^{(j)}(z), P_{k|k}^{(j)}\right) \tag{1.124}$$

$$w_k^{(j)}(z) = \frac{p_{D,k} w_{k|k-1}^{(j)} q_k^{(j)}(z)}{\kappa_k(z) + p_{D,k} \sum_{\ell=1}^{J_{k|k-1}} w_{k|k-1}^{(\ell)} q_k^{(\ell)}(z)} \tag{1.125}$$

$$q_k^{(j)}(z) = \mathcal{N}\left(z; H_k m_{k|k-1}^{(j)}, R_k + H_k P_{k|k-1}^{(j)} H_k^T\right) \tag{1.126}$$

$$m_{k|k}^{(j)}(z) = m_{k|k-1}^{(j)} + K_k^{(j)}\left(z - H_k m_{k|k-1}^{(j)}\right) \tag{1.127}$$

$$P_{k|k}^{(j)} = \left[I - K_k^{(j)} H_k\right] P_{k|k-1}^{(j)} \tag{1.128}$$

$$K_k^{(j)} = P_{k|k-1}^{(j)} H_k^T (H_k P_{k|k-1}^{(j)} H_k^T + R_k)^{-1} \tag{1.129}$$

Each Gaussian in the predicted Gaussian mixture gives rise to $1 + |Z|$ new Gaussians. The new Gaussians receive the same tag identifier from the original Gaussian. Note that Equations (1.120)-(1.121) and (1.127)-(1.129) represent the Kalman Filter prediction and update steps.

4. Pruning:

In this step the Gaussians with low weights are eliminated. Given a trucation threshold T_{th}, the updated Gaussian mixture after pruning of Gaussians with weights $w_k^{(i)} < T_{th}$ is defined as:

$$\tilde{\nu}_k(x) = \sum_{i=1, w_k^{(i)} < T_{th}}^{J_k} \tilde{w}_k^{(i)} \mathcal{N}\left(x; m_k^{(i)}, P_k^{(i)}\right) \tag{1.130}$$

where:

$$\tilde{w}_k^{(i)} = \frac{w_k^{(i)}}{\sum_{j=N_p+1}^{J_k} w_k^{(j)}}$$

5. Target State Estimation:

The individual object state estimate is given by the means of the Gaussians whose associated weights are above a chosen threshold w_{th}:

$$\hat{X}_k = \left\{m_k^{(i)} : w_k^{(i)} > w_{th}\right\} \tag{1.131}$$

and the set of tags associated with the state estimates is given by:

$$\hat{T}_k = \left\{\tau_k^{(i)} : w_k^{(i)} > w_{th}\right\} \tag{1.132}$$

6. Target Trajectories Estimation:

The targets' trajectories and their labels can be extracted according to the following scheme:

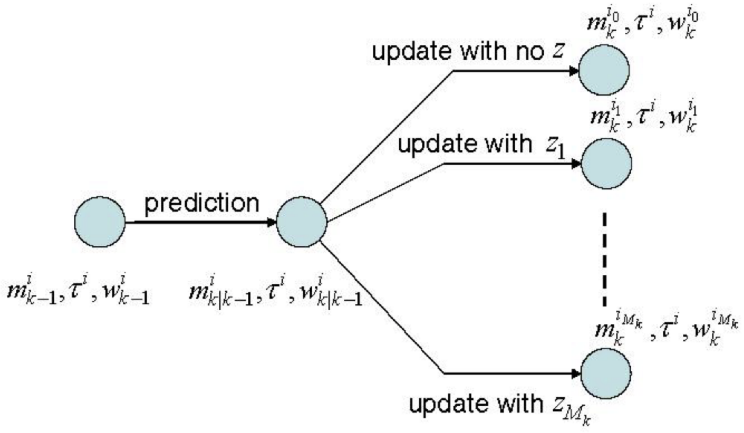

Figure 1.7: A tree structure used to propagate a Gaussian with measurement available at time k. The term $m_k^{i_l}$ denotes the mean of the Gaussian resulting when $m_{k|k-1}^{(i)}$ is updated with measurement z_l, where $l = 0$ represents the case of misdetection. Each Gaussian initiated at time $k = 0$ and new Gaussians from γ_k form the root of a tree that grows linearly with the number of measurements available at each time. Each tree has its unique label that is the same as the tag associated with the Gaussian at the root of the tree. The branches form a possible target trajectory associated with a target over time. Image from [PVC06]

1.3.2 Labeled Random Finite Set (LRFS)

The algorithms based on the standard Random Finite Set (RFS) framework can jointly estimate the number of objects and their states but are unable to estimate target trajectories over time. Filters like the PHD, CPHD, GM-PHD, and SMC-PHD can generate sets of disconnected point estimates at each time, instead of continuous trajectories. The Gaussian-Mixture PHD Tracker (GM-PHDT) try to address the problem of multi-object tracking using some heuristical methods, and it cannot distinguish well between objects that are very close to each other. To overcome this limitation, a Labeled Random Finite Set (LRFS) was introduced by Ba-Tuong Vo and Ba-Ngu Vo [VV13]. The following sections provides a summary of the main techniques utilized in this thesis based on this concept.

In order to distinguish the standard RFS from the labeled RFS it is necessary to define some new notations. Single-object states are denoted by small letters, e.g. $x, \boldsymbol{x}$, while multi-object states are denoted by capital letters, e.g. $X, \boldsymbol{X}$. Spaces are represented by blackboard bold letters, e.g. $\mathbb{X}, \mathbb{Z}, \mathbb{L}, \mathbb{N}$. Following the same idea, labeled states and their distributions (single-object or multi-object) are bolded to differenciate them from the unlabeled representation, e.g. $\boldsymbol{x}, \boldsymbol{X}, \boldsymbol{\pi}$.

A labeled RFS can be defined as a set in which every element $x \in \mathbb{X}$ has been augmented with a label $\ell \in \mathbb{L}$. Suppose an unlabeled set $X = \{x^{(1)}, \cdots, x^{(n)}\}$; a labeled version of this set is obtained by augmenting each $x^{(i)}$ with a label $\ell^{(i)}$:

$$\boldsymbol{X} = \left\{(x^{(1)}, \ell^{(1)}), \cdots, (x^{(n)}, \ell^{(n)})\right\} \tag{1.133}$$

The labeled RFS uses an indicator function to invalidate labeled multi-object states whose labels are not distinct. The indicator function that extracts the labels of a labeled set $\boldsymbol{X}$ is

defined as follows:

$$\mathcal{L}(\boldsymbol{X}) = \{\mathcal{L}(\boldsymbol{x}) : \boldsymbol{x} \in \boldsymbol{X}\} \tag{1.134}$$

If the labeled set $\boldsymbol{X}$ contains multiple elements with the same label, then the cardinality of $\mathcal{L}(\boldsymbol{X})$ will not be equal to the cardinality of $\boldsymbol{X}$. Since by definition, a labeled RFS cannot contain repeated elements, then the following function can be used to determine whether the set $\boldsymbol{X}$ contains repeated labels [VV13]:

$$\Delta(\boldsymbol{X}) = \delta_{|\boldsymbol{X}|}(|\mathcal{L}(\boldsymbol{X})|) \tag{1.135}$$

A labeled RFS with state space $\mathbb{X}$ and discrete label space $\mathbb{L}$ is an RFS on $\mathbb{X}\times\mathbb{L}$ where the labels within each realization are always distinct, i.e., $\Delta(\boldsymbol{X}) = 1$. The unlabeled version of a labeled RFS is defined as the projection of the labeled state space $\mathbb{X} \times \mathbb{L}$ into the unlabeled state space $\mathbb{X}$ applying marginalization [VV13; Mah14]:

$$\pi(\{x^{(1)}, \cdots, x^{(n)}\}) = \sum_{(\ell^{(1)},\cdots,\ell^{(n)})\in\mathbb{L}^n} \boldsymbol{\pi}(\{(x^{(1)}, \ell^{(1)}), \cdots, (x^{(n)}, \ell^{(n)})\})$$

The following subsections introduce the labeled RFS distributions used in this thesis, the Labeled Multi-Bernoulli (LMB) and the Generalized Labeled Multi-Bernoulli (GLMB) [VV13; Reu+14; Mah14].

Labeled Independent Identically Distributed Cluster RFS

The multi-object probability density of a labeled independent and identically distributed (l.i.i.d.) cluster RFS $\boldsymbol{X}$ is defined to be [VV13; Mah14]:

$$\boldsymbol{\pi}(\boldsymbol{X}) = \delta_{\mathbb{L}(n)}(|\{\ell^{(1)}, \cdots, \ell^{(n)}\}|)\rho(|\boldsymbol{X}|)p^{\boldsymbol{X}} \tag{1.136}$$

where $\mathbb{L}(n) = \{\alpha_i \in \mathbb{L}\}_{i=1}^{n}$, $\rho(n)$ is the cardinality probability distribution for $n \geq 0$, and $p(x)$ is the probability density on x representing the spatial distribution of a target object.

If $\boldsymbol{X} = \{(x^{(1)}, \ell^{(1)}), \cdots, (x^{(n)}, \ell^{(n)})\}$ with $|\boldsymbol{X}| = n$, then the above equation can be written as [Mah14]:

$$\boldsymbol{\pi}(\boldsymbol{X}) = \delta_{\{\alpha^{(1)},\cdots,\alpha^{(n)}\}}(\{\ell^{(1)}, \cdots, \ell^{(n)}\})\rho(n)p(x^{(1)})\cdots p(x^{(n)}) \tag{1.137}$$

$$\tag{1.138}$$

$$= \begin{cases} \rho(n)p(x^{(1)})\cdots p(x^{(n)}) & \text{if } \{\alpha^{(1)}, \cdots, \alpha^{(n)}\} = \{\ell^{(1)}, \cdots, \ell^{(n)}\} \\ 0 & \text{otherwise} \end{cases} \tag{1.139}$$

Consequently, for each $n > 1$, $\boldsymbol{\pi}(\{(x^{(1)}, \ell^{(1)}), \cdots, (x^{(n)}, \ell^{(n)})\}) = 0$ for all $(\ell^{(1)}, \cdots, \ell^{(n)})$

exception for the selections given by:

$$\{\ell^{(1)}, \cdots, \ell^{(n)}\} = \{\alpha^{\sigma(1)}, \cdots, \alpha^{\sigma(n)}\} \tag{1.140}$$

where σ is the $n!$ permutations of $1, \cdots, n$. Thus, the set integral $\int \boldsymbol{\pi}(\boldsymbol{X})\delta\boldsymbol{X}$ exists and is finite [Mah14].

Labeled Multi-Bernoulli RFS

A Labeled Multi-Bernoulli (LMB) RFS is a labeled RFS with state space $\mathbb{X}$ and the discrete label space $\mathbb{L}$ distributed according to [VV13; Mah14]:

$$\boldsymbol{\pi}(\{(x^{(1)}, \ell^{(1)}), \cdots, (x^{(n)}, \ell^{(n)})\}) = \delta_n(|\{\ell^{(1)}, \cdots, \ell^{(n)}\}|) \prod_{i \in \mathbb{L}} \left(1 - r^{(i)}\right) \prod_{\ell=1}^{n} 1_{\mathbb{L}}(\ell) \frac{r^{(\ell)} p^{(\ell)}(x)}{1 - r^{(\ell)}} \tag{1.141}$$

where details about the notation can be found in Section 1.2.1 and Appendix A. An LMB RFS is not a multi-Bernoulli RFS on $\mathbb{X} \times \mathbb{L}$, but its unlabeled version is a multi-Bernoulli on $\mathbb{X}$. A compact representation of an LMB density is obtained employing the multi-object exponential notation [VV13]:

$$\pi(\boldsymbol{X}) = \Delta(\boldsymbol{X}) w(\mathcal{L}(\boldsymbol{X})) p^{\boldsymbol{X}} \tag{1.142}$$

where:

$$w(L) = \prod_{i \in \mathbb{L}} \left(1 - r^{(i)}\right) \prod_{\ell \in L} \frac{1_{\mathbb{L}}(\ell) r^{(\ell)}}{1 - r^{(\ell)}} \tag{1.143}$$

$$p(x, \ell) = p^{(\ell)}(x) \tag{1.144}$$

where $r^{(\ell)}$ and $p^{(\ell)}$ are the existence probability and probability density corresponding to label $\ell \in \mathbb{L}$. Thus, an LMB distribution can be represented using the notation $\boldsymbol{\pi}(\boldsymbol{X}) = \{(r^{(\ell)}, p^{(\ell)})\}_{\ell \in \mathbb{L}}$. In addition, the PHD and mean cardinality of the unlabeled LMB RFS are defined by (1.78) and (1.79).

Generalized Labeled Multi-Bernoulli RFS

The Generalized Labeled Multi-Bernoulli (GLMB) is a more sophisticated and flexible class of labeled RFS with state $\mathbb{X}$ and discrete label space $\mathbb{L}$ satisfying the following probability distribution [VV13; Mah14]:

$$\boldsymbol{\pi}(\boldsymbol{X}) = \Delta(\boldsymbol{X}) \sum_{c \in \mathbb{C}} w^{(c)}\left(\mathcal{L}(\boldsymbol{X})\right) \left[p^{(c)}\right]^{\boldsymbol{X}} \tag{1.145}$$

where $\mathbb{C}$ is an arbitrary index set, and enables multiple realizations for a set of track labels $L = \mathcal{L}(\boldsymbol{X})$. The weights $w^{(c)}(L)$ should be normalized, and as well as the spatial distribution $p^{(c)}$, must be a probability density function as follows:

$$\sum_{L\subseteq\mathbb{L}}\sum_{c\in\mathbb{C}} w^{(c)}(L) = 1 \tag{1.146}$$

$$\int_{x\in\mathbb{X}} p^{(c)}(x,\ell)\mathrm{d}x = 1 \tag{1.147}$$

Analyzing (1.142) and (1.145) one can see that the LMB RFS is a particular case of the GLMB RFS with only a single component c for each realization $\boldsymbol{X}$. An LMB RFS encloses statistically independent objects. However, due to the existence of one object affects the association probabilities for other objects, the measurement updated objects are no longer statistically independent. Due to this, an LMB RFS may not precisely represent the multi-object posterior. Unlike the LMB RFS, the GLMB RFS allows multiple components or hypotheses for a set of object labels L, and naturally embodies the data association uncertainty of the measurement update by generating hypothesis c for each possible object to measure association.

δ-Generalized Labeled Multi-Bernoulli RFS

A more specific type of GLMB RFS is the δ-Generalized Labeled Multi-Bernoulli (δ-GLMB) RFS with state $\mathbb{X}$ and discrete label space $\mathbb{L}$ satisfying the following probability distribution:

$$\begin{aligned}\mathbb{C} &= \mathcal{F}(\mathbb{L})\times\Xi \\ w^{(c)}(L) &= w^{(I,\xi)}(L) = w^{(I,\xi)}\delta_I(L) \\ p^{(c)} &= p^{(I,\xi)} = p^{(\xi)}\end{aligned}$$

where Ξ is an arbitrary discrete space, and the distribution of a δ-GLMB RFS can be represented as follows:

$$\boldsymbol{\pi}(\boldsymbol{X}) = \Delta(\boldsymbol{X}) \sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi} w^{(I,\xi)}\delta_I\Big(\mathcal{L}(\boldsymbol{X})\Big)\Big[p^{(I,\xi)}\Big]^{\boldsymbol{X}} \tag{1.148}$$

The sum is taken over by the Cartesian product between the space of finite subsets of $\mathbb{L}$, and the discrete space Ξ. For tracking applications, each subset of $\mathbb{L}$ represents the target labels and Ξ the space of measurement-to-label association histories. An element $(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi$ can be considered a hypothesis that the set of currently existing targets are those with labels I and association history ξ. The weight $w^{(I,\xi)}$ represents the probability of this hypothesis. $p^{(\xi)}(\cdot,\ell)$ is the pdf of the target with label ℓ under association history ξ.

The δ-GLMB representation of an LMB RFS with state space $\mathbb{X}$, discrete finite label space $\mathbb{L}$, and parameter set $\pi = \{r^{(\ell)}, p^{(\ell)}\}_{\ell\in\mathbb{L}}$ can be expressed as [VV13]:

$$\boldsymbol{\pi}(\boldsymbol{X}) = \Delta(\boldsymbol{X}) \sum_{I \in \mathcal{F}(\mathbb{L})} w^{(I)} \delta_I\Big(\mathcal{L}(\boldsymbol{X})\Big) p^{\boldsymbol{X}} \tag{1.149}$$

$$= \Delta(\boldsymbol{X}) w\Big(\mathcal{L}(\boldsymbol{X})\Big) p^{\boldsymbol{X}} \tag{1.150}$$

where the weights $w(L)$ and the spatial distribution $p(x, \ell)$ are defined by (1.143) and (1.144). The Equation 1.150 uses the property that the sums in Equation 1.149 is zero except for $i = \mathcal{L}(\boldsymbol{X})$. Additionally, (1.150) is equivalent to the LMB density (1.142).

1.3.3 Labeled multi-object dynamic model

Algorithms based on the Labeled Random Finite Set (LRFS) assume a standard multi-object transition model with the inclusion of target labels. Given the current multi-object state $\boldsymbol{X}$, every existing target $(x, \ell) \in \boldsymbol{X}$ has probability $P_S(x, \ell)$ of surviving to the next time and evolves to a new state (x_+, ℓ_+) with probability density $f(x_+|x, \ell)\delta_\ell(\ell+)$, or dies with probability $1 - P_S(x, \ell)$. The set of surviving objects at the next time is distributed according to:

$$f_S\left(\boldsymbol{S}|\boldsymbol{X}\right) = \Delta(\boldsymbol{S})\, \Delta(\boldsymbol{X})\, 1_{\mathcal{L}(\boldsymbol{X})}\left(\mathcal{L}(\boldsymbol{S})\right) [\Phi(\boldsymbol{S}; \cdot)]^{\boldsymbol{X}} \tag{1.151}$$

where:

$$\Phi(\boldsymbol{S}; x, \ell)] = \sum_{(x_+, \ell_+) \in \boldsymbol{S}} \delta_\ell(\ell_+) P_S(x, \ell) f(x_+|x, \ell) + \left(1 - 1_{\mathcal{L}(\boldsymbol{S})}\right)\left(1 - P_S(x, \ell)\right) \tag{1.152}$$

and the set of new objects born at the next time step is distributed as follows:

$$f_B(\boldsymbol{B}) = \Delta(\boldsymbol{B}) w_B(\mathcal{L}(\boldsymbol{B}))[P_B(\cdot)]^{\boldsymbol{B}} \tag{1.153}$$

where $P_B(\cdot, \ell)$ is the single object target birth density associated with label ℓ, and $w_B(\cdot)$ is the birth weight. The multi-object state at the next time $\boldsymbol{X}_+$ is the union of surviving and newborn targets. Assuming that targets' states evolve independently of each other, and that birth targets are independent of surviving targets, the multi-object transition can be expressed in the following form [VV13]:

$$f(\boldsymbol{X}_+|\boldsymbol{X}) = f_S(\boldsymbol{X}_+ \cap (\mathbb{X} \times \mathbb{L})|\boldsymbol{X})\, f_B(\boldsymbol{X}_+ - (\mathbb{X} \times \mathbb{L})) \tag{1.154}$$

1.3.4 Labeled multi-object observation model

For a given multi-object state at time k $\boldsymbol{X}$, each state $(x, \ell) \in \boldsymbol{X}$ has probability $P_D(\boldsymbol{x})$ of generating a detection z with likelihood $g(z|x)$, and probability $1 - P_D(\boldsymbol{x})$ of being misdetected. Assuming that the set of detected objects W with elements are conditionally independent

and W is a multi-Bernoulli RFS with parameter set $\{(P_D(\boldsymbol{x}), g(\cdot|\boldsymbol{x})) : \boldsymbol{x} \in \boldsymbol{X}\}$, it is possible to represent the probability density of W as:

$$\pi_D(W|\boldsymbol{X}) = \Big\{(P_D(\boldsymbol{x}), g(\cdot|\boldsymbol{x})) : \boldsymbol{x} \in \boldsymbol{X}\Big\}(W) \tag{1.155}$$

Considering K the set of clutter observations independent of target detections, and modeled by a Poisson RFS with intensity $\kappa(\cdot)$, its distribution can be represented accordingly:

$$\pi_K(K) = e^{-\langle \kappa, 1\rangle} \kappa^K \tag{1.156}$$

The multi-object observation $Z = \{z_1, \cdots, z_{|z|}\}$ is the union of the detected objects and Poisson clutter with intensity κ as follows:

$$g(Z|\boldsymbol{X}) = e^{-\langle \kappa, 1\rangle} \kappa \sum_{\theta \in \Theta} \Big[\Psi_Z(\cdot; \theta)\Big]^{\boldsymbol{X}} \tag{1.157}$$

where:

$$\Psi_Z(x, \ell; \theta) = \begin{cases} \dfrac{P_D(x, \ell) g(z_{\theta(\ell)}|x, \ell)}{\kappa(z_{\theta(\ell)})} & \theta(\ell) > 0 \\ 1 - P_D(x, \ell) & \theta(\ell) = 0 \end{cases} \tag{1.158}$$

The set Θ is called the association map space, where $\theta : \mathbb{L} \to \{0, 1, \cdots, |Z|\}$, such that $[\theta(i) = \theta(j) > 0] \Rightarrow [i = j]$.

1.3.5 Labeled Optimal Subpattern Assignment (LOSPA) metric

Performance evaluation of multi-object tracking algorithms is necessary for the design, parameter optimization and comparison of tracking systems. The goal of performance evaluation is to measure the distance between two sets of objects: the ground truth objects and the set of estimated objects. The Optimal SubPattern Assignment (OSPA) metric is defined on the space of finite sets of vectors and is inadequate for the evaluation of the labeled RFS [Ris+11].

The OSPA metric evaluates the differences with respect to the number of objects and the residual between the estimated and the true object states for each time k. Consequently, the OSPA does not incorporate information about track continuity or switching of track IDs. To overcome this drawback, Ristic et al. proposed the Labeled Optimal SubPattern Assignment (LOSPA), which extends the OSPA metric by incorporating labeling information in the distance calculation. The distance of two labeled state vectors $\boldsymbol{x} = (x, \ell)$ and $\boldsymbol{y} = (y, s)$ in the LOSPA is defined by:

$$d(\boldsymbol{x}, \boldsymbol{y}) = \Big(d(x, y)^p + \Big(\alpha(1 - \delta_\ell(s))\Big)^p\Big)^{\frac{1}{p}} \tag{1.159}$$

where ℓ and s are the labels of the objects. The parameter $\alpha \in [0, c]$ influences the penalty due to labeling errors. Chosing $\alpha = 0$, makes the second term of (1.159) disappears, and consequently the LOSPA distance becomes equivalent to the OSPA distance. When $\alpha = c$, the LOSPA distance of two objects with labels $\ell \neq s$ is $d(\boldsymbol{x}, \boldsymbol{y}) = c$, and can be interpreted as the mismatching labels are penalized like a missed detection [Ris+11].

1.3.6 The Generalized Labeled Multi-Bernoulli (GLMB) filter

The Generalized Labeled Multi-Bernoulli (GLMB) filter is the first implementable and provably Bayes-optimal multi-target tracking algorithm [Mah14; Mah18]. In the GLMB filter, targets are labeled by an ordered pair of intergers $\ell = (k, i)$, where k is the time of birth, and i is a unique index that distinguish objects born at the same time. An existing target at time k has state $(\boldsymbol{x}, \ell)$ consisting of the kinematic state $\boldsymbol{x} \in \mathbb{X}$ and label $\ell \in \mathbb{L}_{0:k}$. To ensure that the labels are distinct, it's required that X and the set of labels of X have the same cardinality. The function $\Delta(X) \overset{\text{def}}{=} \delta_{|X|}(|\mathcal{L}|)$, called the distinct label indicator, can be used to verify the cardinality.

The assignement of labels to measurements at time k is an association mapping $\theta : \mathbb{L}_{0:k} \to \{0, 1, \cdots, |Z|\}$ such that $\theta(\ell) = \theta(\ell') > 0$ implies $\ell = \ell'$. The set of all association maps at time k is denoted Θ_k. In the GLMB filter, the multi-object filtering density at time $k-1$ is a GLMB of the form:

$$\pi_{k-1}(X|Z_{0:k-1}) = \Delta(X) \sum_{\xi \in \Theta_{0:k-1}} w_{k-1}^{(\xi)}(\mathcal{L}(X))[p_{k-1}^{(\xi)}]^X \tag{1.160}$$

The cardinality distribution of the GLMB is given by:

$$\rho_{k-1}(n) = \sum_{L \in \mathcal{F}(\mathbb{L})} \sum_{\xi \in \Theta_{0:k-1}} \delta_n(|L|) w_{k-1}^{(\xi)}(L) \tag{1.161}$$

Each $\xi = (\theta_0, \cdots, \theta_{k-1}) \in \Theta_{0:k-1}$ represents a history of association maps up to time $k-1$, and contains the history of target labels encapsulating object births and deaths.

The GLMB density is a conjugate prior with respect to the standard multi-object likelihood function and is also closed under the multi-object prediction. Under the standard multi-object model, and if the multi-object density π_{k-1} at time $k-1$ is a GLMB as defined above, then the multi-object prediction density $\pi_{k|k-1}$ is a GLMB as follows [VV13; Mah14]:

$$\pi_{k|k-1}(X|Z_{0:k-1}) = \Delta(X) \sum_{\xi \in \Theta_{0:k-1}} w_{k|k-1}^{(\xi)}(\mathcal{L}(X))[p_{k|k-1}^{(\xi)}]^X \tag{1.162}$$

where

$$w^{(\xi)}_{k|k-1}(L) = w^{(\xi)}_{S,k|k-1}(L \cap \mathbb{L}_{0:k-1}) w_{\Gamma,k}(L \cap \mathbb{L}_k) \tag{1.163}$$

$$p^{(\xi)}_{k|k-1}(x,\ell) = 1_{\mathbb{L}_{0:k-1}}(\ell) p^{(\xi)}_{S,k|k-1}(x,\ell) + 1_{\mathbb{L}_k}(\ell) p_{\Gamma,k}(x,\ell) \tag{1.164}$$

$$w^{(\xi)}_{S,k|k-1}(L) = \left[\bar{P}^{(\xi)}_{S,k|k-1}\right]^L \sum_{I \supseteq L} \left[1 - \bar{P}^{(\xi)}_{S,k|k-1}\right]^{I-L} w^{(\xi)}_{k-1}(I) \tag{1.165}$$

$$\bar{P}^{(\xi)}_{S,k|k-1}(\ell) = \langle P_{S,k|k-1}(\cdot,\ell), p^{(\xi)}_{k-1}(\cdot,\ell) \rangle \tag{1.166}$$

$$p^{(\xi)}_{S,k|k-1}(x,\ell) = \frac{\langle P_{S,k|k-1}(\cdot,\ell) f_{k|k-1}(\boldsymbol{x}|\cdot,\ell), p^{(\xi)}_{k-1}(\cdot,\ell) \rangle}{\bar{P}^{(\xi)}_{S,k|k-1}(\ell)} \tag{1.167}$$

and the multi-object filtering density π_k is a GLMB given by:

$$\pi_k(X|Z_{0:k}) = \Delta(X) \sum_{\xi \in \Theta_{0:k-1}} \sum_{\theta \in \Theta_k} w^{(\xi,\theta)}_k(\mathcal{L}(X)|Z_k) \left[p^{(\xi,\theta)}(\cdot|Z_k)\right]^X \tag{1.168}$$

where

$$w^{(\xi,\theta)}_k(L|Z_k) \propto 1_{\Theta_k(L)}(\theta) \left[\bar{\Psi}^{(\xi,\theta)}_{Z,k}\right]^L w^{(\xi)}_{k|k-1}(L) \tag{1.169}$$

$$\bar{\Psi}^{(\xi,\theta)}_{Z,k}(\ell) = \langle \Psi^{(\theta)}_{Z,k}(\cdot,\ell), p^{(\xi)}_{k|k-1}(\cdot,\ell) \rangle \tag{1.170}$$

$$\Psi^{(\theta)}_{Z,k}(x,\ell) = \begin{cases} \dfrac{\psi_k(z_{\theta(\ell)};(x,\ell))}{\lambda_{F,k}} & \text{if} \quad \theta(\ell) > 0 \\ 1 - P_{D,k}(x,\ell) & \text{if} \quad \theta(\ell) = 0 \end{cases} \tag{1.171}$$

$$p^{(\xi,\theta)}_k(x,\ell|Z) = \frac{\Psi^{(\theta)}_{Z,k}(x,\ell) p^{(\xi)}_{k|k-1}(x,\ell)}{\bar{\Psi}^{(\xi,\theta)}_{Z,k}(\ell)} \tag{1.172}$$

The above recursion is the first exact closed-form solution to the Bayes multi-object filter [Mah14; Mah18; VV13]. Due to the growing number of components at each iteration, truncating the GLMB sum is needed. Vo, Vo, and Phung proposed an implementation of the GLMB filter that discards components with weak weights. This truncation also minimizes the L1 error of the multi-object density, and it has a cubic complexity in the number of observations in the worst case [Mah14; Pap+14]. Another well-known approximation of the GLMB filter is the Labeled Multi-Bernoulli (LMB) filter [Reu+14], which is the labeled analog of an unlabeled Poisson RFS, and, consequently, the LMB filter can be interpreted as a Labeled Probability Hypothesis Density (LPHD) filter for the standard multi-object measurement model [Mah17]. This insight is employed in this thesis to derive and implement a new filter: the LPHD filter.

1.4 Conclusion

This chapter presented the main features of the Bayesian estimation for single and multi-object tracking, followed by the Random Finite Set (RFS) and labeled RFS formalism used

for the derivation of the new filters proposed in Chapter 2 and 3. Numerous filtering methods such as the PHD filters, the Kalman filter, the Extended Kalman filter (EKF), the Unscented Kalman filter (UKF), and the Particle filters were presented. Two metrics that will be applied to evaluate the performance of the tracking algorithms were introduced. Finally, we introduced the PHD Tracker (PHDT) and the Generalized Labeled Multi-Bernoulli (GLMB) filters that serve as a basis for the development of the new filters proposed in this thesis.

Chapter 2

The labeled version of the PHD filter

This chapter introduces the labeled version of the PHD filter to address the multi-object tracking problem within the Labeled Random Finite Set (LRFS). The difference between the standard RFS and the labeled RFS is that the standard RFS filters produce only multiple state estimates (multi-object filtering) while the LRFS filters produce multiple object trajectory estimates (multi-object tracking). The first implementations of RFS filters did not address multi-object tracking because of computational concerns, and some solutions proposed to solve the multi-object tracking and labeling problem were in general based on heuristics. The labeled RFS theory of Ba-Tuong Vo and Ba-Ngu Vo, introduced in Section 1.3.2, was the first rigorous theoretical formulation of true multi-object tracking [VV13; Mah14; VVP14]. Their proposal led to the Generalized Labeled Multi-Bernoulli (GLMB) filter, which is the first computationally tractable and provably Bayes-optimal multi-object tracking algorithm (see Section 1.3.6).

The original version of the GLMB filter is almost intractable, and many approximations were proposed. Ba-Ngu Vo, Ba-Tuong, and Hoang proposed a fast implementation of the GLMB filter with joint prediction and update that has computational complexity $O(n^2m)$, where n is the current number of objects and m the current number of measurements [HVV15; VVH17]. This efficient implementation has average run time at least two orders of magnitude faster than the original algorithm. Thus, the purpose of this chapter is to answer the following questions: if an analogous labeled version of the PHD filter exists, what is its relationship with the GLMB filter? Is it a suboptimal but computationally faster alternative?

In this chapter, we will show that an approximation of the GLMB filter, known as the Labeled Multi-Bernoulli (LMB) filter [Reu+14], can be reinterpreted as a rigorous theoretical labeled PHD filter with the following form:

$$\cdots \rightarrow \quad \nu_{k-1}(x,\ell) \quad \xrightarrow{\text{prediction}} \quad \nu_{k|k-1}(x,\ell) \quad \xrightarrow{\text{update}} \quad \nu_{k|k}(x,\ell) \rightarrow \cdots$$

and according to Mahler [Mah14; Mah17]:

$$\nu_{k-1}(x,\ell) = r_{k-1}^{(\ell)} p_{k-1}(x,\ell) = r_{k-1}^{(\ell)} p_{k-1}^{(\ell)}(x) \tag{2.1}$$

represents a labeled PHD, where $0 \leq r_{k-1}^{(\ell)} \leq 1$ and $p_{k-1}^{(\ell)}(x)$ are the probability of existence and the spatial distribution of the object with label ℓ.

This chapter addresses the derivation and implementation of a labeled version of the Probability Hypothesis Density (PHD) filter and is organized as follows: Section 2.2 presents a sumary of heuristic labeled PHD approaches. The Labeled Multi-Bernoulli (LMB) filter necessary to the labeled PHD derivation is introduced in Section 2.3. Then, the Labeled PHD filter and its implementation using Gaussian-mixture method is proposed as a reinterpretation of the LMB filter.

2.1 Labeled PHD notations and abbreviations

Throughout this chapter, we use the notations and abbreviations presented in Sections 1.2.1, 1.3.2, and Appendix A. We also use the Probability Generating Functional (PGFl) to prove the propositions necessary for the derivation of the LPHD filter. The following sections presents some complementary definitions.

Let $\boldsymbol{X}$ be an RFS of the space $\mathbb{X}$, and $\boldsymbol{\pi}(\boldsymbol{X})$ its probability distribution. Then its PGFl is defined by:

$$G[h] = \int h^{\boldsymbol{X}} \boldsymbol{\pi}(\boldsymbol{X}) \delta \boldsymbol{X} \tag{2.2}$$

where for all test functions $0 \leq h(x, \ell) \leq 1$ and $h^{\boldsymbol{X}}$ is given by:

$$h^{\boldsymbol{X}} = \begin{cases} 1 & \text{if } \boldsymbol{X} = \emptyset \\ \prod\limits_{(x,\ell)\in\boldsymbol{X}} h(x,\ell) & \text{otherwise} \end{cases} \tag{2.3}$$

It is possible to reconstruct the distribution $\boldsymbol{\pi}(\boldsymbol{X})$ from the PGFl $G[h]$ as follows [Mah14]:

$$\boldsymbol{\pi}(\boldsymbol{X}) = \frac{\delta G}{\delta \boldsymbol{X}}[0] \tag{2.4}$$

where $\delta/\delta\boldsymbol{X}$ represents an iterated functional derivative. For $\boldsymbol{X} = \{(x, \ell)\}$, the functional derivative can be heuristically determined by:

$$\frac{\delta G}{\delta(x,\ell)}[h] = \lim_{\epsilon\to 0} \frac{G[h + \epsilon\delta_{(x,\ell)} - G[h]}{\epsilon} \tag{2.5}$$

where $\delta_{(x,\ell)}(x', \ell') = \delta_\ell(\ell')\delta_x(x')$ is the Dirac delta function on $\boldsymbol{X}$ concentrated at (x, ℓ).

2.2 Heuristic labeled PHD filter

The first implementations of the PHD filter presented in Section 1.2.7 appeared in 2003 and were based on the particle methods. Due to computational complexity, track labeling was ignored. The solutions presented were based on heuristics such as nearest-neighbor to address the problem of joint estimation of the number of tracks and their states.

The Gaussian-Mixture PHD (GM-PHD) filter introduced in Section 1.3.1 presented an essential role in the development of PHD filters with labels. A label is assigned to each Gaussian component, and simple heuristics rules allow the propagation of the labels for the prediction and update steps [CPV06; PVC06; Mah14]. However, tracking performance tends to degrade when the clutter is dense. The PHD implementations based on this idea can be interpreted as an approximation of the Labeled Random Finite Set (LRFS) [Mah17]:

$$\boldsymbol{\pi}(\{(x^{(1)},\ell^{(1)}),\cdots,(x^{(n)},\ell^{(n)})\}) \cong 0 \tag{2.6}$$

where $(\ell^{(1)},\cdots,\ell^{(n)})$ are not distinct, and since it is an approximation, the requirement that the labels are distinct can be ignored. In the real world a target cannot simultaneously have different kinematic states, so that is why the labels should be distinct. This heuristic PHD with label implementation substitutes the state x with $\boldsymbol{x} = (x,\ell)$ whenever x occurs. The naïve labeled PHD filter has the following form [Mah14; Mah17]:

$$\nu_{k|k-1}(x,\ell) = \beta_{k|k-1}(x,\ell) + \sum_{\ell'}\int p_S(x',\ell')f_{k|k-1}(x,\ell|x',\ell')\nu_{k-1}(x',\ell')\mathrm{d}x' \tag{2.7}$$

$$\nu_{k|k}(x,\ell) = \left(1 - p_D(x,\ell) + \sum_{z\in Z_k}\frac{p_D(x,\ell)g_k(z|x,\ell)}{\kappa_k(z)+\tau_k(z)}\right)\nu_{k|k-1}(x,\ell) \tag{2.8}$$

$$\tau_k(z) = \sum_{\ell}\int p_D(x,\ell)g_k(z|x,\ell)\nu_{k|k-1}(x,\ell)\mathrm{d}x \tag{2.9}$$

where $f_{k|k-1}(x,\ell|x',\ell') = \delta_\ell(\ell')f_{k|k-1}(x,x')$. The labeled PHD $\nu^{(\ell)}_{k|k}(x) = \nu_{k|k}(x,\ell)$ is equivalent to propagating multiple kinematic PHDs in parallel, one for each label ℓ. Additionally, the assumption in (2.6) if $(\ell^{(1)},\cdots,\ell^{(n)})$ are not distinct breaks down when target tracks or clutter are very close. In this situation, this naïve labeled PHD filter tends to perfom poorly. Formal definition of LRFS theory is therefore necessary.

2.3 The Labeled Multi-Bernoulli (LMB) filter

The GLMB filter, introduced in Section 1.3.6, keeps a history of association maps between tracks and measurements, which makes the number of posterior components grow exponentially. The δ-GLMB filter was proposed to reduce both the computation and the need to store a huge history of associations by approximating the number of hypotheses using a direct truncation of the multi-object posterior [VV13]. Tractable techniques for truncating the posterior and prediction densities were proposed based on the k-shortest paths and ranked assignment algorithms [Mur68; Yen71; Epp98].

The Labeled Multi-Bernoulli (LMB) filter proposed by Reuter et al. is an approximation of the δ-Generalized Labeled Multi-Bernoulli (δ-GLMB) filter [Reu+14]. The LMB filter propagates the predicted and posterior multi-object densities using an LMB RFS rather than a GLMB RFS or a δ-GLMB RFS. The main advantages are that the LMB approximation significantly reduces the computational cost in scenarios with a large number of objects, and that its number of components grows linearly whereas the growth of the δ-GLMB filter is exponential.

The multi-object posterior of the LMB filter is assumed to follow an LMB RFS (see Section 1.3.2) with state space $\mathbb{X}$ and label space $\mathbb{L}$ given as [Reu+14]:

$$\pi(\boldsymbol{X}) = \Delta(\boldsymbol{X})w(\mathcal{L}(\boldsymbol{X}))p^{\boldsymbol{X}} \tag{2.10}$$

where:

$$w(L) = \prod_{i\in\mathbb{L}}\left(1 - r^{(i)}\right)\prod_{\ell\in L}\frac{1_{\mathbb{L}}(\ell)r^{(\ell)}}{1 - r^{(\ell)}} \tag{2.11}$$

$$p(x,\ell) = p^{(\ell)}(x) \tag{2.12}$$

Additionally, the multi-object birth density of an LMB RFS with state space $\mathbb{X}$ and label space $\mathbb{B}$ has the following form:

$$\pi_B(\boldsymbol{X}) = \Delta(\boldsymbol{X})w_B(\mathcal{L}(\boldsymbol{X}))[p_B]^{\boldsymbol{X}} \tag{2.13}$$

with:

$$w_B(L) = \prod_{i\in\mathbb{B}}\left(1 - r_B^{(i)}\right)\prod_{\ell\in L}\frac{1_{\mathbb{B}}(\ell)r_B^{(\ell)}}{1 - r_B^{(\ell)}} \tag{2.14}$$

$$p_B(x,\ell) = p_B^{(\ell)}(x) \tag{2.15}$$

where the labels $\ell \in \mathbb{B}$ of the newborn objects must be distinct and the sets $\mathbb{L}$ and $\mathbb{B}$ have to be disjoint to ensure unique labels, i.e., $\mathbb{L} \cap \mathbb{B} = \emptyset$.

2.3.1 LMB prediction

Considering that an LMB RFS is a special case of a GLMB RFS, the prediction of an LMB RFS can be expressed by a GLMB RFS [Reu+14; Mah14] as follows:

$$\boldsymbol{\pi}_+(\boldsymbol{X}_+) = \Delta(\boldsymbol{X}_+)w_+(\mathcal{L}(\boldsymbol{X}_+))[p_+]^{\boldsymbol{X}_+} \tag{2.16}$$

where the symbol $_+$ is used to express the predicted values, and the time index notation $k-1$ and $k|k-1$ are suppressed to maintain the simplicity. The state space $\mathbb{L}_+ = \mathbb{L} \cup \mathbb{B}$, the predicted weights w_+ and the predicted spatial distribution p_+ are:

$$w_+(I_+) = w_B(I_+ \cap \mathbb{B}) w_S(I_+ \cap \mathbb{L}) \tag{2.17}$$

$$p_+(x,\ell) = 1_{\mathbb{L}}(\ell) p_{+,S}(x,\ell) + 1_{\mathbb{B}}(\ell) p_B(x,\ell) \tag{2.18}$$

$$p_{+,S}(x,\ell) = \frac{\langle p_S(\cdot,\ell) f_+(x|\cdot,\ell), p(\cdot,\ell)\rangle}{\eta_S(\ell)} \tag{2.19}$$

$$\eta_S(\ell) = \int \langle p_S(\cdot,\ell) f_+(x|\cdot,\ell), p(\cdot,\ell)\rangle \mathrm{d}x \tag{2.20}$$

$$w_S(L) = \left[\eta_S\right]^L \sum_{I \supseteq L} \left[1-\eta_S\right]^{I-L} w(I) \tag{2.21}$$

where I_+ denotes a set of track labels, $p_S(x,\ell)$ is the state dependent survival probability, $f_+(x|\cdot,\ell)$ is the single-object Markov transition density. The term $w_S(L)$ denotes the probability that all tracks with label $\ell \in L$ of a superset $I \supseteq L$ survive while all objects with label $\ell \notin L$ disappear. The above equations correspond to a prediction of a δ-GLMB RFS, but due to the LMB prior, the history of association maps ξ is missing. As the LMB RFS is a special case of the GLMB RFS which comprises only a single component c (see Section 1.3.2), the sum over all hypotheses (I,ξ) in (1.148) reduces to the single term in (2.16).

The predicted multi-object density (2.16) is the GLMB representation of the LMB RFS [Reu+14]:

$$\boldsymbol{\pi}_+ = \left\{\left(r_+^{(\ell)}, p_+^{(\ell)}\right)\right\}_{\ell\in\mathbb{L}_+} = \left\{\left(r_{+,S}^{(\ell)}, p_{+,S}^{(\ell)}\right)\right\}_{\ell\in\mathbb{L}} \bigcup \left\{\left(r_{+,B}^{(\ell)}, p_{+,B}^{(\ell)}\right)\right\}_{\ell\in\mathbb{B}} \tag{2.22}$$

with:

$$r_{+,S}^{(\ell)} = \eta_S^{(\ell)} r^{(\ell)} \tag{2.23}$$

$$p_{+,S}^{(\ell)} = \langle p_S(\cdot,\ell) f_+(x|\cdot,\ell), p(\cdot,\ell)\rangle / \eta_S^{(\ell)} \tag{2.24}$$

where $p_S(x,\ell)$ is the state dependent survival probability, $f_+(x|\cdot,\ell)$ is the single-object Markov transition density, and $\eta_S(\ell)$ is given by (2.20).

2.3.2 LMB update

Suppose that the multi-object predicted density is an LMB RFS with state space $\mathbb{X}$, finite label space $\mathbb{L}_+$, and parameter set $\boldsymbol{\pi}_+ = \{(r_+^{(\ell)}, p_+^{(\ell)})\}_{\ell\in\mathbb{L}_+}$ as defined in (2.22). The updated LMB RFS, given a set of observation Z, is expressed as [Reu+14]:

$$\boldsymbol{\pi}(\boldsymbol{X}|Z) = \left\{\left(r^{(\ell)}, p^{(\ell)}\right)\right\}_{\ell\in\mathbb{L}_+} \tag{2.25}$$

with parameters:

$$r^{(\ell)} = \sum_{(I+,\theta)\in\mathcal{F}(\mathbb{L}_+)\times\Theta_{I_+}} w^{(I+,\theta)}(Z)1_{I_+}(\ell) \tag{2.26}$$

$$p^{(\ell)} = \frac{1}{r^{(\ell)}} \sum_{(I+,\theta)\in\mathcal{F}(\mathbb{L}_+)\times\Theta_{I_+}} w^{(I+,\theta)}(Z)1_{I_+}(\ell)p^{(\theta)}(x,\ell) \tag{2.27}$$

and:

$$w^{(I+,\theta)}(Z) = \frac{\delta_{\theta^{-1}(\{0:|Z|\})}(I_+)w_+(I_+)[\eta_Z^{(\theta)}]^{I_+}}{\sum_{(I+,\theta)\in\mathcal{F}(\mathbb{L}_+)\times\Theta_{I_+}} \delta_{\theta^{-1}(\{0:|Z|\})}(I_+)w_+(I_+)[\eta_Z^{(\theta)}]^{I_+}} \tag{2.28}$$

$$p^{(\theta)}(x,\ell|Z) = \frac{p_+(x,\ell)\psi_Z(x,\ell;\theta)}{\eta_Z^{(\theta)}(\ell)} \tag{2.29}$$

$$\eta_Z^{(\theta)}(\ell) = \langle p_+(\cdot,\ell), \psi_Z(\cdot,\ell;\theta)\rangle \tag{2.30}$$

$$\psi_Z(x,\ell;\theta) = \delta_0(\theta(\ell))q_D(x,\ell) + \Big(1-\delta_0(\theta(\ell))\Big)\frac{p_D(x,\ell)g(z_)\theta(\ell)|x,\ell)}{\kappa(z_{\theta(\ell)})} \tag{2.31}$$

where I_+ denotes a set of track labels and Θ_{I_+} represents the space of mappings $\theta : I_+ \to \{0,\cdots,|Z|\}$ in a manner that $\theta(i) = \theta(i') > 0$ infers $i = i'$. The term $p_D(x,\ell)$ is the detection probability of the track, $q_D(x,\ell) = 1-p_D(x,\ell)$ is the probability of missed detection, $g(z_{\theta(\ell)}|x,\ell)$ is the single likelihood of the measurement $z_{\theta(\ell)}$ given a track with label ℓ, and $\kappa(z) = \lambda_c c(z)$ represents the intensity of Poisson clutter.

2.4 The Labeled Probability Hypothesis Density (LPHD) filter

This section introduces the Labeled Probability Hypothesis Density (LPHD) filter. Using the Labeled Random Finite Set (LRFS) to derive a labeled PHD would be done by approximating an evolving labeled RFS as a labeled independent and identically distributed (l.i.i.d.) cluster as defined in Section 1.3.2. However, this approach can be problematic; as considered in (1.136), the only possible realizations of $\mathcal{L}(\boldsymbol{X})$ are $\mathbb{L}(n) = \{\alpha^{(1)},\cdots,\alpha^{(n)}\}$ for $n \geq 0$. The impasse is that the objects with labels $(\alpha^{(1)},\cdots,\alpha^{(n)})$ are affirmed to exist with certainty, which is rarely the case in real scenarios. Targets may disappear, in which case their labels also disappear. What, then, should be done alternatively?

Proposition 1. *Considering that the PHD of a Multi-Bernoulli RFS, defined in (1.78), has the form $\nu(x) = \sum_{j=1}^{M} r^{(j)}p^{(j)}(x)$, a labeled PHD of an Labeled Multi-Bernoulli (LMB) RFS with state space $\mathbb{X}$ and discrete label space $\mathbb{L}$ can be defined by:*

$$\nu(x',\ell') = \sum_{\ell\in\mathbb{L}} \delta_\ell(\ell')r^{(\ell)}p^{(\ell)}(x') \tag{2.32}$$

where $0 \leq r^{(\ell)} \leq 1$ and $p^{(\ell)}(x)$ are, respectively, the probability of existence and the spatial distribution of the object with label ℓ.

Proof. Considering the behavior that a multi-object tracker exhibits, for each time k there is a total number of labels represented by v_k in a way that a set of labels assigned to tracks at that time has the following form:

$$\mathbb{L}_{k|k} = \mathcal{L}(v_k) = \{\alpha^{(1)}, \cdots, \alpha^{(v_k)}\} \tag{2.33}$$

If $\nu_{k|k}(x, \ell)$ is the labeled PHD at time k, then the total number of expected targets at time k is:

$$N_{k|k} = \sum_{\ell \in \mathbb{L}_{k|k}} \int \nu_{k|k}(x, \ell) \mathrm{d}x = \sum_{\ell \in \mathbb{L}_{k|k}} r_{k|k}^{(\ell)} \tag{2.34}$$

where the expected number of targets that have label ℓ, i.e., the probability of existence, can be expressed as:

$$r_{k|k}^{(\ell)} = \int \nu_{k|k}(x, \ell) \mathrm{d}x \qquad 0 \leq r_{k|k}^{(\ell)} \leq 1 \tag{2.35}$$

Because the multi-object probability density of an LMB RFS (see Equation 2.10) is a mono-GLMB distribution, $w(L)$ has the form:

$$w(L)^J = \prod_{\ell \in J - L} \left(1 - r^{(\ell)}\right) \prod_{\ell \in L} \frac{1_L(\ell) r^{(\ell)}}{1 - r^{(\ell)}} \tag{2.36}$$

for some $J \subseteq \mathbb{L}$ with $0 \leq r^{(\ell)} \leq 1$ for all $\ell \in J$. According to Mahler, its PGFl is defined by [Mah14; Mah17]:

$$G_{k|k}[h] = \prod_{\ell \in \mathbb{L}_{k|k}} (1 - r_{k|k}^{(\ell)} + r_{k|k}^{(\ell)} p_{k|k}^{(\ell)}[h]) \tag{2.37}$$

where $p_{k|k}^{(\ell)}[h] = \int p_{k|k}(x, \ell) \mathrm{d}x$ with the spatial distribution of an object with label ℓ is represented by:

$$p_{k|k}(x, \ell) = \frac{\nu_{k|k}(x, \ell)}{r_{k|k}^{(\ell)}} \tag{2.38}$$

Then the PHD of $G_{k|k}[h]$ is given by [Mah14; Mah17]:

$$\nu_{k|k}(x',\ell') = \left[\frac{\delta G_{k|k}}{\delta(x',\ell')}[h]\right]_{h=1} \tag{2.39}$$

$$= \left[\left(\prod_{\ell\in\mathbb{L}_{k|k}} (1 - r_{k|k}^{(\ell)} + r_{k|k}^{(\ell)} p_{k|k}^{(\ell)}[h])\right)\left(\sum_{\ell\in\mathbb{L}_{k|k}} \frac{\delta_\ell(\ell') r_{k|k}^{(\ell)} p_{k|k}^{(\ell)}(x')}{1 - r_{k|k}^{(\ell)} + r_{k|k}^{(\ell)} p_{k|k}^{(\ell)}[h]}\right)\right]_{h=1} \tag{2.40}$$

$$= \sum_{\ell\in\mathbb{L}_{k|k}} \delta_\ell(\ell') r_{k|k}^{(\ell)} p_{k|k}^{(\ell)}(x') = r_{k|k}^{(\ell')} p_{k|k}(x',\ell') \tag{2.41}$$

Thus, an LMB RFS is completely described by its labeled PHD $\nu(x,\ell) = r^{(\ell)}p(x,\ell)$, i.e., it is a labeled analog of an unlabeled Poisson RFS. ■

We conclude that the Labeled Multi-Bernoulli (LMB) filter proposed by Reuter et al. can be reinterpreted as a Labeled Probability Hypothesis Density (LPHD) filter. Additionally, Equation 2.37 shows that any LRFS can be approximated as an LMB RFS, which has the same labeled PHD [Mah14; Mah17]. The following subsections discuss the expression of the prediction and update steps of the LPHD filter.

2.4.1 LPHD prediction

The concept behind the Labeled Probability Hypothesis Density (LPHD) filter is the representation of the predicted and posterior multi-object densities using an LMB RFS, which ensures the same advantages of the LMB filter.

Proposition 2. *The prediction equation for the LPHD filter is the same as that for the LMB filter. Thus, the prediction step of the LPHD filter can be expressed by:*

$$\nu_+(x,\ell) = \begin{cases} \int p_S(x',\ell)\, f_+(x|x')\, \nu(x',\ell)\mathrm{d}x & \text{if } \ell \in \mathbb{L} \\ \nu^B(x,\ell) & \text{if } \ell \in \mathbb{B} \end{cases} \tag{2.42}$$

where $\mathbb{B} \subseteq \mathbb{L}_+$ is the labeled set for the birth targets, $\mathbb{L}$ is the labeled set for the current targets, and $\nu^B(x,\ell) = r_B^{(\ell)} p_B^{(\ell)}(x,\ell)$ is the birth target LPHD.

Proof. Defining the multi-object target birth and the posterior PGFls as follows:

$$G_B[h] = \left(\prod_{\ell\in\mathbb{B}} (1 - r_B^{(\ell)} + r_B^{(\ell)} p_B^{(\ell)}[h])\right) \tag{2.43}$$

$$G[h] = \left(\prod_{\ell\in\mathbb{L}} (1 - r^{(\ell)} + r^{(\ell)} p^{(\ell)}[h])\right) \tag{2.44}$$

where the labels $\ell \in \mathbb{B}$ of the newborn objects must be distinct and the sets $\mathbb{L}$ and $\mathbb{B}$ have to be disjoint to ensure unique labels, i.e., $\mathbb{L} \cap \mathbb{B} = \emptyset$. Note that to maintain the simplicity,

time index notations $k-1$ and $k|k-1$ are suppressed. Using the symbol $_+$ to express the predicted values, and defining the predicted state space as $\mathbb{L}_+ = \mathbb{L} \cup \mathbb{B}$, the predicted PGFl can be expressed by:

$$G_+[h] = \prod_{\ell \in \mathbb{L}_+} \left(1 - r_+^{(\ell)} + r_+^{(\ell)} p_+^{(\ell)}[h]\right) \tag{2.45}$$

where:

$$r_+^{(\ell)} = \begin{cases} r_B^{(\ell)} & \text{if } \ell \in \mathbb{B} \\ r^{(\ell)} p^{(\ell)}[p_S] & \text{if } \ell \in \mathbb{L} \end{cases} \tag{2.46}$$

$$p_+^{(\ell)}(x) = \begin{cases} p_B^{(\ell)}(x) & \text{if } \ell \in \mathbb{B} \\ \dfrac{\int p^{(\ell)}(x') p_S(x') f_+(x|x') \mathrm{d}x'}{p^{(\ell)}[p_S]} & \text{if } \ell \in \mathbb{L} \end{cases} \tag{2.47}$$

■

2.4.2 LPHD update

Consider the predicted LPHD $\nu_+(x,\ell)$ and PGFl $G_+[h]$:

$$\nu_+(x,\ell) = 1_J(\ell) r_+^{(\ell)} p_+^{(\ell)}(x) = 1_J(\ell) r_+^{(\ell)} p_+(x,\ell)$$
$$G_+[h] = \prod_{\ell \in \mathbb{L}_+} \left(1 - r_+^{(\ell)} + r_+^{(\ell)} p_+^{(\ell)}[h]\right)$$

Proposition 3. *The update equations for the LPHD filter is the same as that for the LMB filter. Thus, the update step of the LPHD filter can be expressed as:*

$$\nu(x',\ell') = L_Z(x',\ell') p_+(x',\ell') \tag{2.48}$$

where $L_Z(x',\ell')$ is the application-dependent measurement noise model.

Proof. Suppose a Poisson clutter with PGFl $G[g] = e^{\kappa[g-1]}$, where $\kappa[g] = \int g(z) \cdot \kappa(z) \mathrm{d}z$. Let Z be the new measurement set, and the predicted PGFl given by:

$$G_+[h] = \prod_{\ell \in \mathbb{L}_+} \left(1 - r_+^{(\ell)} + r_+^{(\ell)} p_+^{(\ell)}[h]\right) \tag{2.49}$$

and defining R_α as:

$$R_\alpha = \left(\prod_{\ell \in \mathbb{L}_\alpha^c} \left(1 - r_+^{(\ell)} p_+^{(\ell)}[p_D]\right)\right) \left(\prod_{\ell \in \mathbb{L}_\alpha} \frac{r_+^{(\ell)} p_+^{(\ell)}[p_D L_{Z_{\alpha(\ell)}}]}{\kappa(Z_{\alpha(\ell)})}\right) \tag{2.50}$$

According to Mahler [Mah17], the measurement-update PGFl is given by:

$$G[h] = \frac{\sum_\alpha \left(\prod_{\ell \in \mathbb{L}_\alpha^c} \left(1 - r_+^{(\ell)} + r_+^{(\ell)} p_+^{(\ell)}[h(1-p_D)]\right)\right)\left(\prod_{\ell \in \mathbb{L}_\alpha} \frac{r_+^{(\ell)} p_+^{(\ell)}[h p_D L_{Z_{\alpha(\ell)}}]}{\kappa(Z_{\alpha(\ell)})}\right)}{\sum_\alpha \left(\prod_{\ell \in \mathbb{L}_\alpha^c} \left(1 - r_+^{(\ell)} p_+^{(\ell)}[p_D]\right)\right)\left(\prod_{\ell \in \mathbb{L}_\alpha} \frac{r_+^{(\ell)} p_+^{(\ell)}[p_D L_{Z_{\alpha(\ell)}}]}{\kappa(Z_{\alpha(\ell)})}\right)} \tag{2.51}$$

where $\mathbb{L}_\alpha = \{\ell \in \mathbb{L}_+ | \alpha(\ell) > 0\}$, and $\alpha : \mathbb{L}_+ \to \{0, \cdots, |Z|\}$ is an association $\alpha(\ell) = \alpha(\ell') > 0$ that means that $\ell = \ell'$. The summations are performed over all associations α. With these elements, we can derive the measurement-updated PHD as follows:

$$\nu(x', \ell') = \frac{\delta G}{\delta(x', \ell')}[1] = L_Z(x', \ell') p_+(x', \ell') \tag{2.52}$$

where:

$$L_Z(x', \ell') = \frac{\sum_\alpha R_\alpha \left(\frac{1_{\mathbb{L}_\alpha^c}(\ell') r_+^{(\ell')}(1 - p_D(x', \ell'))}{1 - r_+^{(\ell')} p_+^{(\ell')}[p_D]} + \frac{1_{\mathbb{L}_\alpha}(\ell') p_D(x', \ell')) L_{Z_{\alpha(\ell')}}(x', \ell')}{p_+^{(\ell')}[p_D L_{Z_{\alpha(\ell')}}]} \right)}{\sum_\alpha R_\alpha} \tag{2.53}$$

where the updated value for $r^{(\ell')}$ can be obtained from (2.35):

$$r^{(\ell')} = \frac{\sum_\alpha R_\alpha \left(\frac{1_{\mathbb{L}_\alpha^c}(\ell') r_+^{(\ell')} p_+^{(\ell')}[1 - p_D]}{1 - r_+^{(\ell')} p_+^{(\ell')}[p_D]} + 1_{\mathbb{L}_\alpha}(\ell') \right)}{\sum_\alpha R_\alpha} \tag{2.54}$$

and the updated value for $p(x', \ell')$ can be obtained from (2.38):

$$p(x, \ell') = \hat{L}_Z(x', \ell') p_+(x', \ell') \tag{2.55}$$

where:

$$\hat{L}_Z(x', \ell') = \frac{\sum_\alpha R_\alpha \left(\frac{1_{\mathbb{L}_\alpha^c}(\ell') r_+^{(\ell')}(1 - p_D(x', \ell'))}{1 - r_+^{(\ell')} p_+^{(\ell')}[p_D]} + \frac{1_{\mathbb{L}_\alpha}(\ell') p_D(x', \ell')) L_{Z_{\alpha(\ell')}}(x', \ell')}{p_+^{(\ell')}[p_D L_{Z_{\alpha(\ell')}}]} \right)}{\sum_\alpha R_\alpha \frac{1_{\mathbb{L}_\alpha^c}(\ell') r_+^{(\ell')} p_+^{(\ell')}[1 - p_D]}{1 - r_+^{(\ell')} p_+^{(\ell')}[p_D]} + 1_{\mathbb{L}_\alpha}(\ell')} \tag{2.56}$$

■

Additionally, an LMB RFS is a special case of the GLMB RFS and the δ-GLMB RFS (see Section 1.3.2). Using the property defined in (1.142), the predicted LMB RFS can be equivalently expressed in the δ-GLMB form see (2.22):

$$\boldsymbol{\pi}_+(\boldsymbol{X}_+) = \Delta(\boldsymbol{X}_+) \sum_{I_+ \in \mathcal{F}(\mathbb{L}_+)} w_+(I_+)\delta_{I_+}\Big(\mathcal{L}(\boldsymbol{X}_+)\Big)\Big[p_+\Big]^{\boldsymbol{X}_+} \tag{2.57}$$

The next step is to calculate the δ-GLMB update. Due to the LMB tracks' representation, the history of association maps is no longer available in the above equation. Thus, the δ-GLMB update can be expressed by [VV13]:

$$\boldsymbol{\pi}(\boldsymbol{X}|Z) = \Delta(\boldsymbol{X}) \sum_{(I_+,\theta) \in \mathcal{F}(\mathbb{L}_+)\times\Theta_{I_+}} w^{(I_+,\theta)}(Z)\delta_{I_+}\Big(\mathcal{L}(\boldsymbol{X})\Big)\Big[p^{(\theta)}(\cdot|Z)\Big]^{\boldsymbol{X}} \tag{2.58}$$

where $w^{(I_+,\theta)}(Z)$ is the measurement updated weights defined in (2.28) and $p^{(\theta)}(\cdot|Z)$ is the spatial distribution defined in (2.29). The cardinality distribution of a δ-GLMB RFS is obtained by [VV13]:

$$\rho(n) = \sum_{(I,\theta)\in\mathcal{F}(\mathbb{L})\times\Theta} \sum_{L\in\mathcal{F}_n(\mathbb{L})} w^{(I,\theta)}\delta_I\Big(L\Big) = \sum_{(I,\theta)\in\mathcal{F}_n(\mathbb{L})\times\Theta} w^{(I,\theta)} \tag{2.59}$$

Then the probability of cardinality n is calculated with the summation of the weights of all Hypotheses (I,θ) with $|I| = n$. The δ-GLMB representation makes it possible to extract information about a track label ℓ using its PHD:

$$\begin{aligned}\nu(x) &= \sum_{(I_+,\theta)\in\mathcal{F}(\mathbb{L}_+)\times\Theta} \sum_{\ell\in\mathbb{L}_+} p^{(\theta)}(x,\ell) \sum_{L\subseteq\mathbb{L}_+} 1_L(\ell) w^{(I,\theta)}(Z)\delta_{I_+}(L) &(2.60)\\ &= \sum_{\ell\in\mathbb{L}_+} \sum_{(I_+,\theta)\in\mathcal{F}(\mathbb{L}_+)\times\Theta} w^{(I,\theta)}(Z) 1_{I_+}(\ell) p^{(\theta)}(x,\ell) &(2.61)\end{aligned}$$

with the PHD of track label ℓ given by:

$$\nu^{(\ell)}(x) = \sum_{(I_+,\theta)\in\mathcal{F}(\mathbb{L}_+)\times\Theta} w^{(I,\theta)}(Z) 1_{I_+}(\ell) p^{(\theta)}(x,\ell) \tag{2.62}$$

The integral over the PHD of a track with label ℓ can be interpreted as the existence probability of the track ℓ. Thus, the track's existence probability is:

$$r^{(\ell)} = \langle \nu^{(\ell)}, 1\rangle = \sum_{(I_+,\theta)\in\mathcal{F}(\mathbb{L}_+)\times\Theta} w^{(I,\theta)}(Z) 1_{I_+}(\ell) \tag{2.63}$$

and its spatial distribution, calculated by normalization, is:

$$p^{(\ell)}(x) = \frac{\nu^{(\ell)}(x)}{\langle \nu^{(\ell)}, 1\rangle} = \frac{1}{r^{(\ell)}} \sum_{(I_+,\theta)\in\mathcal{F}(\mathbb{L}_+)\times\Theta} w^{(I,\theta)}(Z) 1_{I_+}(\ell) p^{(\theta)}(x,\ell) \tag{2.64}$$

According to (1.78), the PHD of (2.25) is given by:

$$\nu^{\mathrm{LMB}}(x) = \sum_{\ell\in\mathbb{L}_+} r^{(\ell)} p^{(\ell)}(x) = \sum_{\ell\in\mathbb{L}_+} \langle \nu^{(\ell)}, 1\rangle \frac{\nu^{(\ell)}(x)}{\langle \nu^{(\ell)}, 1\rangle} = \sum_{\ell\in\mathbb{L}_+} \nu^{(\ell)}(x) \tag{2.65}$$

2.5 The Gaussian-Mixture LPHD (GM-LPHD) filter

The proposed Labeled Probability Hypothesis Density (LPHD) filter is a reinterpretation of the Labeled Multi-Bernoulli (LMB) filter with a few modifications. Compared to the Probability Hypothesis Density (PHD) filter, the LPHD filter is computationally more expensive, but more accurate with no cardinality bias. Although, the LPHD filter significantly reduces the computational complexity of the prediction step. Similar to the LMB filter, the representation of the predicted LMB RFS in δ-GLMB form is still computationally expensive and requires many hypotheses in scenarios with a large number of objects.

The following subsections provide full details of the LPHD filter implementation as well as explicit equations for the Gaussian-mixture (GM) implementation: Gaussian-Mixture Labeled PHD (GM-LPHD).

2.5.1 GM-LPHD prediction

The prediction step of the LPHD filter directly implements the result of Equation 2.39 and Section 2.4.1. If the multi-object posterior is an LMB RFS with label space $\mathbb{L}$, its LPHD and parameter set is defined by:

$$\nu(x', \ell') = \sum_{\ell\in\mathbb{L}} \delta_{\ell}(\ell') r^{(\ell)} p^{(\ell)}(x') = r^{(\ell')} p(x', \ell')$$

$$\boldsymbol{\pi} = \left\{ \left(r^{(\ell)}, p^{(\ell)} \right) \right\}_{\ell\in\mathbb{L}}$$

thus, the predicted LPHD with the LMB RFS distribution with label space $\mathbb{L}_+ = \mathbb{L} \cup \mathbb{B}$ is given by:

$$\nu_+(x, \ell) = \begin{cases} \int p_S(x', \ell)\, f_+(x|x')\, \nu(x', \ell)\mathrm{d}x & \text{if } \ell \in \mathbb{L} \\ \nu^B(x, \ell) & \text{if } \ell \in \mathbb{B} \end{cases}$$

$$\boldsymbol{\pi}_+ = \left\{ \left(r_{+,S}^{(\ell)}, p_{+,S}^{(\ell)} \right) \right\}_{\ell\in\mathbb{L}} \bigcup \left\{ \left(r_B^{(\ell)}, p_B^{(\ell)} \right) \right\}_{\ell\in\mathbb{B}}$$

where $\mathbb{B} \subseteq \mathbb{L}_+$ is the labeled-set for the birth targets, $\mathbb{L}$ is the labeled set for the surviving targets, and $\nu^B(x, \ell) = r_B^{(\ell)} p_B^{(\ell)}(x, \ell)$ is the birth target LPHD. The labels for the newborn objects should be distinct. The posterior probability densities $p^{(\ell)}$ of all tracks $\ell \in \mathbb{L}$ are given by a mixture of Gaussians as follows:

$$p^{(\ell)}(x) = \sum_{j=1}^{j^{(\ell)}} w^{(\ell,j)} \mathcal{N}\left(x; \hat{x}^{(\ell,j)}, P^{(\ell,j)}\right)$$

where $\hat{x}^{(\ell,j)}$ is the mean value of each Gaussian, and $P^{(\ell,j)}$ is the estimation error covariance. Similar to the LMB filter, the weights $w^{(\ell,j)}$ should be normalized. The hypothesis is that each object follows a linear Gaussian process model with a state independent survival probability, i.e., $p_S(x), p_S$, a state transition matrix F, and a process noise covariance Q. The linear Gaussian process model can be expressed by:

$$f_+(x|\xi) = \mathcal{N}\left(x; F\xi, Q\right) \tag{2.66}$$

Thus, the predicted existence probability $r_{+,S}^{(\ell)}$ and the predicted spatial distribution $p_{+,S}^{(\ell)}(x)$ of a survival object ℓ are given by:

$$r_{+,S}^{(\ell)} = r^{(\ell)} p_S \tag{2.67}$$

$$p_{+,S}^{(\ell)}(x) = \sum_{j=1}^{j^{(\ell)}} w^{(\ell,j)} \mathcal{N}\left(x; \hat{x}_+^{(\ell,j)}, P_+^{(\ell,j)}\right) \tag{2.68}$$

where:

$$\hat{x}_+^{(\ell,j)} = F\hat{x}^{(\ell,j)} \tag{2.69}$$

$$P_+^{(\ell,j)} = FP^{(\ell,j)}F^T + Q \tag{2.70}$$

where each of j^ℓ Gaussian distributions of an object ℓ is predicted using the standard Kalman filter. For nonlinear problems, it is possible to use the Extended Kalman filter (EKF) or the Unscented Kalman filter (UKF). The prediction equations are analogous to those of the Kalman filter; see Section 1.1.2 for further detail.

The number of newborn objects and their existence probability are application dependent. For a birth model with number of Gaussian components J^B, object states $\hat{x}_B^{\ell,j}$, and estimation error covariance $P_B^{(\ell,j)}$, the spatial distribution of the newborn Bernoulli object is given by:

$$p_{+,B}^{(\ell)} = \sum_{j=1}^{J_B} w_B^{(\ell,j)} \ \mathcal{N}\left(x; \hat{x}_B^{(\ell,j)}, P_B^{(\ell,j)}\right) \tag{2.71}$$

where the expected number of newborn objects is given by $\hat{N}_B = \sum_{\ell \in \mathbb{B}} r_B^{(\ell)}$.

2.5.2 GM-LPHD update

In order to calculate the posterior existence probability and the posterior spatial distribution of individual objects defined in Section 2.4.2, the update step of the LPHD filter needs the

innovations of each object ℓ with all observations $z_{\theta(\ell)}$.

This section proposes explicit forms for the GM implementation of the likelihoods $\eta_Z^{(\theta)}(\ell)$ and the measurement updated posterior distributions $p^{(\theta)}(x,\ell|Z)$ defined in (2.30) and (2.29). Given the predicted PHD:

$$\nu_+(x,\ell) = 1_J(\ell) r_+^{(\ell)} p_+^{(\ell)}(x) = 1_J(\ell) r_+^{(\ell)} p_+(x,\ell)$$

with the predicted spatial distribution of an object ℓ is a Gaussian-mixture given by:

$$p_+(x,\ell)(x) = \sum_{j=1}^{J_+^{(\ell)}} w_+^{(\ell,j)} \mathcal{N}\left(x; \hat{x}_+^{(\ell,j)}, P_+^{(\ell,j)}\right) \tag{2.72}$$

Similar to the PHD filter (see Section 1.2.8) and the LMB filter, the detection probability is assumed state independent, i.e., $p_D(x) = p_D$. To simplify notations, the measurement matrix H and the measurement noise R are assumed to be constant. The likelihood of the assignment of observations $z_{\theta(\ell)}$ to object ℓ is defined as folllows:

$$\eta_Z^{(\theta)}(\ell) = \frac{p_D}{\kappa(z_{\theta(\ell)})} \sum_{j=1}^{J_+^{(\ell)}} w_+^{(\ell,j)} \mathcal{N}\left(z_{\theta(\ell)}; z_+^{(\ell,j)}, S_+^{(\ell,j)}\right) \tag{2.73}$$

where $\kappa(\cdot)$ is the intensity of the false alarm process that follows a Poisson distribution. The innovation covariance $S_+^{(\ell,j)}$ and the predicted measurement $z_+^{(\ell,j)}$ are calculated using the Kalman filter equations:

$$S_+^{(\ell,j)} = H P_+^{(\ell,j)} H^T + R \tag{2.74}$$

$$z_+^{(\ell,j)} = H_k \hat{x}_+^{(\ell,j)} \tag{2.75}$$

The posterior spatial distribution of the update of object ℓ with observation $z_+^{(\ell,j)}$ is defined by:

$$p^{(\theta)}(x,\ell|Z) = \sum_{j=1}^{J_+^{(\ell)}} w_+^{(\ell,j,\theta)}(Z) \mathcal{N}\left(x; \hat{x}_+^{(\ell,j,\theta)}, P^{(\ell,j)}\right) \tag{2.76}$$

where the posterior weight of each Gaussian component j has the following form:

$$w^{(\ell,j,\theta)}(Z) = \frac{\frac{1}{\kappa(z_{\theta(\ell)})}\, p_D\, w_+^{(\ell,j)} \mathcal{N}\left(z_{\theta(\ell)}; z_+^{(\ell,j)}, S^{(\ell,j)}\right)}{\eta_Z^{(\theta)}(\ell)} \tag{2.77}$$

with:

$$\hat{x}_{+}^{(\ell,j,\theta)}(Z) = \hat{x}_{+}^{(\ell,j)} + K^{(\ell,j)}\left(z_{\theta(\ell)} - z_{+}^{(\ell,j)}\right) \tag{2.78}$$

$$K^{(\ell,j)} = P_{+}^{(\ell,j)} H^T \left[S^{(\ell,j)}\right]^{-1} \tag{2.79}$$

$$P^{(\ell,j)} = P_{+}^{(\ell,j)} - K^{(\ell,j)} S^{(\ell,j)} \left[K^{(\ell,j)}\right]^T \tag{2.80}$$

The proposed measurement update equations can be extended for nonlinear problems using the EKF or the UKF filter. The missed detection probability $\eta_Z^{(\theta)}(\ell) = q_D$, where $q_D = 1 - p_D$, corresponds to a missed detection of an object ℓ, i.e., $\theta(\ell) = 0$. In the case of the probability of missed detection is state independent, the posterior spatial distribution of the track corresponds to the predicted spatial distribution in the following form:

$$p^{(\theta)}(x, \ell | Z) = \sum_{j=1}^{J_{+}^{(\ell)}} w_{+}^{(\ell,j,\theta)}(Z) \mathcal{N}\left(x; \hat{x}_{+}^{(\ell,j,\theta)}, P^{(\ell,j)}\right) \tag{2.81}$$

where:

$$w^{(\ell,j,\theta)}(Z) = w_{+}^{(\ell,j)} \tag{2.82}$$

$$\hat{x}^{(\ell,j,\theta)}(Z) = \hat{x}_{+}^{(\ell,j)} \tag{2.83}$$

$$P^{(\ell,j)} = P_{+}^{(\ell,j)} \tag{2.84}$$

2.5.3 GM-LPHD track extraction and prunning

In general, algorithms for target extraction estimate the number of objects $\hat{N}$ using the Maximum a posteriori (MAP) estimate of the cardinality distribution. Similar to the LMB filter, the track extraction in the LPHD filter is not restricted to the MAP estimate of the cardinality distribution approach. Since the LPHD filter represents its posterior distribution as an LMB RFS, it is possible to perform the track extraction selecting all tracks whose existence probability exceeds a threshold T. Thus, the set of extracted tracks can be calculated by:

$$\hat{X} = \{(\hat{x}, \ell) | r^{(\ell)} > T\} \tag{2.85}$$

The threshold T is application dependent. A high value for T increases the delay for including newborn tracks in the output of the tracking system and decreases the number of false tracks. Smaller values for T increases the number of false tracks and decreases the output delay for newborn tracks.

In the case where the $p_D \approx 1$, a missed detection considerably reduces the existence probability, what might suppress the output of a previously confirmed track with $r^{(\ell)} \approx 1$. To mitigate this issue, a hysteresis can be used to return outputs only if the maximum existence probability $r_{\max}^{(\ell)}$ of a track ℓ has once exceeded an upper threshold T_{upper} and if

the current existence probability $r^{(\ell)}$ is higher than a lower threshold T_{lower}:

$$\hat{X} = \{(\hat{x}, \ell) : r_{\max}^{(\ell)} > T_{\text{upper}} \wedge r^{(\ell)} > T_{\text{lower}}\} \tag{2.86}$$

2.6 Numerical simulation using the LPHD filter

In this section, a nonlinear Gaussian scenario is presented in order to demonstrate and verify the performance of the proposed Labeled Probability Hypothesis Density (LPHD) filter. The LPHD filter is implemented in the Gaussian-mixture form, using an Extended Kalman filter (EKF) to perform the update step. A performance comparison is also presented with the δ-GLMB filter and the PHD Tracker (PHDT) filter (see Section 1.3.1).

The nonlinear scenario is the same used by Ba-Ngu Vo, Ba-Tuong, and Reuter et al. to evaluate the GLMB filter and the LMB filter [VV13; Reu+14]. It counts with a time-varying number of objects observed in clutter, i.e., the scenario comprises a total of ten objects that appear and disappear at different times. Figure 2.1 shows the trajectories of the objects with the start and stop positions of each track.

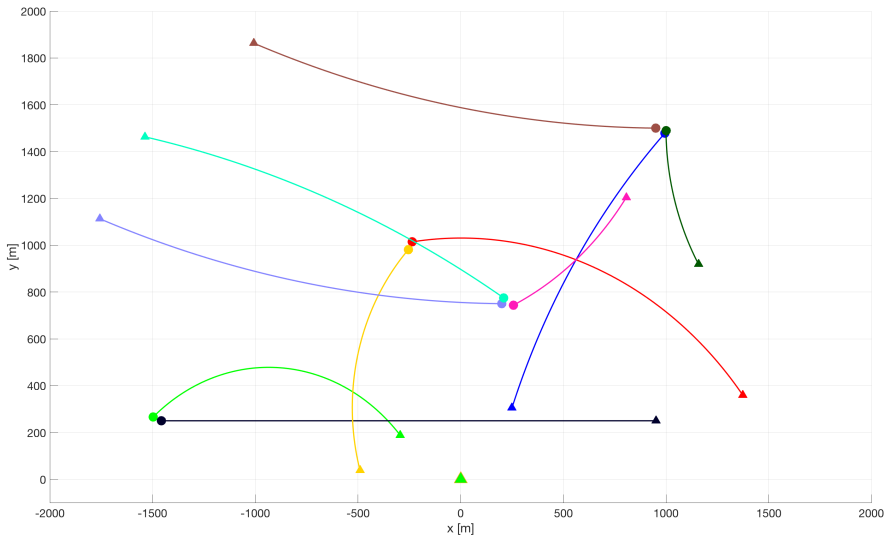

Figure 2.1: Trajectories of ten time-varying observed targets during $100s$. The number of objects varies over time, and the simulation starts with one object, followed by three new objects that appear at time $10s$; one new object arises at time $20s$ followed by three other objects that appear at time $40s$; finally, two objects appear at time $60s$. The radar is at the origin of the axes. Start and stop positions for each track are shown with • and ▲.

The object states are represented by a vector $x = [P_x, V_x, P_y, V_y, \omega]^T$ that contains the positions (P_x, P_y), the velocities (V_x, V_y), and the turn rate ω. The transition turn model used in this example is defined by:

$$\begin{aligned} \tilde{x}_k &= F(\omega_{k-1})\tilde{x}_{k-1} + Gw_{k-1} \\ \omega_k &= \omega_{k-1} + \Delta u_{k-1} \end{aligned}$$

where:

$$F(\omega) = \begin{bmatrix} 1 & \frac{\sin\omega\Delta}{\omega} & 0 & -\frac{1-\cos\omega\Delta}{\omega} \\ 0 & \cos\omega\Delta & 0 & -\sin\omega\Delta \\ 0 & \frac{1-\cos\omega\Delta}{\omega} & 1 & \frac{\sin\omega\Delta}{\omega} \\ 0 & \sin\omega\Delta & 0 & \cos\omega\Delta \end{bmatrix}, \qquad G = \begin{bmatrix} \frac{\Delta^2}{2} & 0 \\ T & 0 \\ 0 & \frac{\Delta^2}{2} \\ 0 & \Delta \end{bmatrix}$$

with $w_{k-1} \sim \mathcal{N}(\cdot; 0, \sigma_w^2 I)$, $u_{k-1} \sim \mathcal{N}(\cdot; 0, \sigma_u^2 I)$, I is the identity matrix, $\Delta = 1s$, $\sigma_w = 15m/s^2$, and $\sigma_u = (\pi/180)\text{rad}/s$. The probability of survival objects is $p_S(x) = 0.99$, and when the object is detected, the observation is a noise bearing range defined by:

$$z_k = \begin{bmatrix} \arctan(\frac{P_x}{P_y}) \\ \sqrt{P_x^2 + P_y^2} \end{bmatrix} + \epsilon_k \tag{2.87}$$

where $\epsilon_k \sim \mathcal{N}(\cdot; 0, R_k)$, with $R_k = \text{diag}([\sigma_\theta^2, \sigma_r^2]^T)$, $\sigma_\theta = (\pi/180)$ rad, and $\sigma_r = 5m$. The birth objects may only appear at four different locations, which are modeled by a multi-Bernoulli RFS:

$$\pi_B = \left\{ \left(r_B^{(i)}, p_B^{(i)} \right) \right\}_{i=1}^{4}$$

where the spatial distribution of the birth objects is Gaussian $p_B^{(i)} \sim \mathcal{N}(x; \hat{x}_B^{(i)}, P_B)$. The existence probability $r_B^{(i)}$ of the Bernoulli distributions, the mean values $\hat{x}_B^{(i)}$ of the Gaussian distributions, and the covariance of Gaussian birth terms P_B are given by:

$$\begin{aligned} r_B^{(1)} &= r_B^{(2)} = 0.02 \\ r_B^{(3)} &= r_B^{(4)} = 0.03 \end{aligned}$$

$$\begin{aligned} \hat{x}_B^{(1)} &= [-1500, 0, 250, 0, 0]^T \\ \hat{x}_B^{(2)} &= [-250, 0, 1000, 0, 0]^T \\ \hat{x}_B^{(3)} &= [250, 0, 750, 0, 0]^T \\ \hat{x}_B^{(4)} &= [1000, 0, 1500, 0, 0]^T \end{aligned}$$

$$P_B = \text{diag}([50, 50, 50, 50, 6(\pi/180)]^T)^2$$

The probability of detection is state-dependent and reaches a peak value of 0.98 at the origin and 0.92 at the edge of the surveillance region. The probability of detection can be calculated as follows:

$$p_{D,k}(x) = \frac{0.98\mathcal{N}([P_{x,k}, P_{y,k}]^T; 0, 6000^2 I_2)}{\mathcal{N}(0; 0, 6000^2 I_2)}$$

where I_2 denotes the identity matrix of dimension two. Additionally, clutter follows a Poisson RFS with intensity given by:

$$\kappa(z) = \frac{\lambda_c}{V}\mathcal{U}(z)$$

where $\mathcal{U}(z)$ denotes a uniform density in the observation region, and $V = \int \mathcal{U}(z)\mathrm{d}z$. For this example we set $\lambda_c = 15$ false alarms per scan.

The LPHD filter output for a single run is depicted in Figure 2.2, showing the true and estimated tracks in x and y coordinates versus time. The filter produces accurate estimates of individual target states with a small delay. No track switching is observed, assuring that the estimated track identities are consistent for the entire scenario. However, track label switching can occur during target crossings, as is expected of any tracking algorithm due to the high process noise of motion model, and the presence of false alarms, missed detections, and clutter.

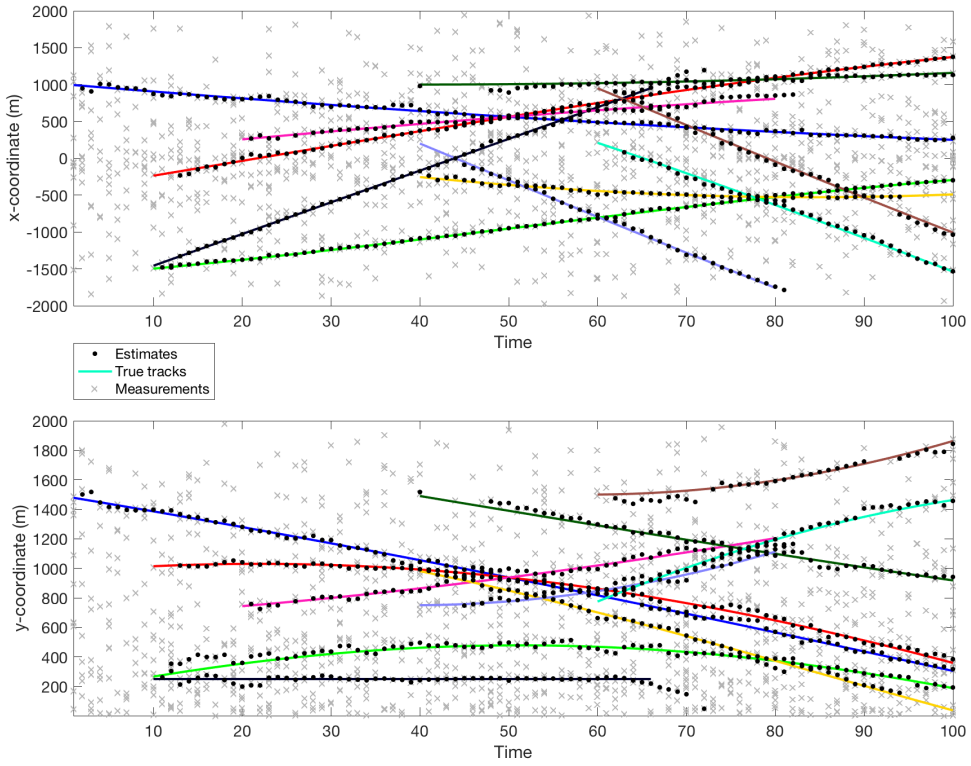

Figure 2.2: Tracks and estimates for x and y coordinates.

The performance of the proposed filter is compared to the δ-GLMB filter and the PHDT filter. For the prediction and update steps, the δ-GLMB keeps 1000 components with the highest weights selected using the k-shortest paths and the Murthy's algorithm [Mur68]. For the LPHD and δ-GLMB, the components with weights $w < 10^{-5}$ are discarded. The number of new components calculated and stored in each propagation is defined as proportional to the weight of the original component such that at least 100 terms are kept to represent each cardinality hypothesis, and the entire density is truncated to a maximum of 1000 terms. For

the PHDT filter, gating is performed at each time step using a 99% validation gate. Pruning and merging are performed at each time step using a weight threshold of $T_{th} = 10^{-5}$ and a merging threshold of $U = 4m$ (see Section 1.3.1).

We compare the performance of the proposed algorithm by running a Gaussian-mixture (GM) implementation of the PHDT and the δ-GLMB filters with 100 Sequential Monte Carlo (SMC) trials. Figure 2.3 shows the cardinality statistics averaged over 100 SMC runs for all the filters. The results confirm that, for this specific scenario and set-up, the LPHD and the δ-GLMB produce similar results, outperforming the PHDT filter. The poor performance of the PHDT filter is associated with the maner in which this filter estimates the cardinality distribution in scenarios where targets appear and disappear.

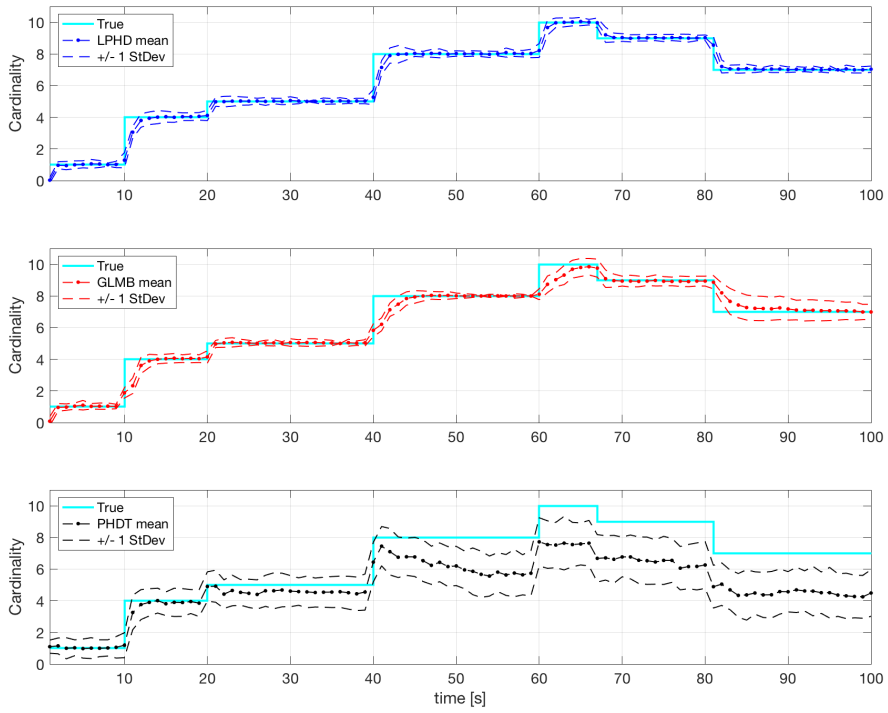

Figure 2.3: Cardinality statistics for the LPHD, δ-GLMB, and PHDT filters with clutter rate $\lambda_c = 15$ and $p_D = 0.98$, averaged over 100 Monte Carlo runs.

Figure 2.4 compares the Optimal SubPattern Assignment (OSPA) metric, with $p = 1$ and $c = 100m$ (see Section 1.2.9). As expected, the LPHD filter provides smaller OSPA distance than the PHDT filter. However, the computational complexity of the LPHD filter update using the labeled RFS is significantly higher than the PHDT filter that implements the standard RFS. The OSPA distance for the LPHD filter is similar to the δ-GLMB filter. However, the OSPA metric can be influenced by the different algorithms' setups.

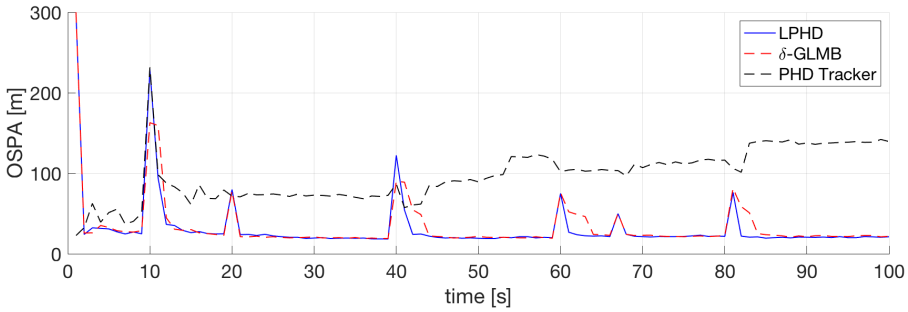

Figure 2.4: OSPA distance ($p = 1, c = 300m$) for LPHD, δ-GLMB, and PHDT filters with clutter rate $\lambda_c = 15$ and $p_D = 0.98$, averaged over 100 Monte Carlo runs.

Figure 2.5 and 2.6 show the OSPA distance and the cardinality statistics for a similar simulation with a lower detection probability, $p_D = 0.75$, and a lower clutter rate, $\lambda_c = 10$ false alarms per scan. Compared to Figure 2.3 and 2.4, the OSPA distance and the cardinality variance are slightly higher, but the performance of the filters are still very similar. This numerical simulation presented the pratical application of the proposed algorithm; further studies and rigorous performance analysis are beyond the scope of this chapter, and will need to be the subject of future research.

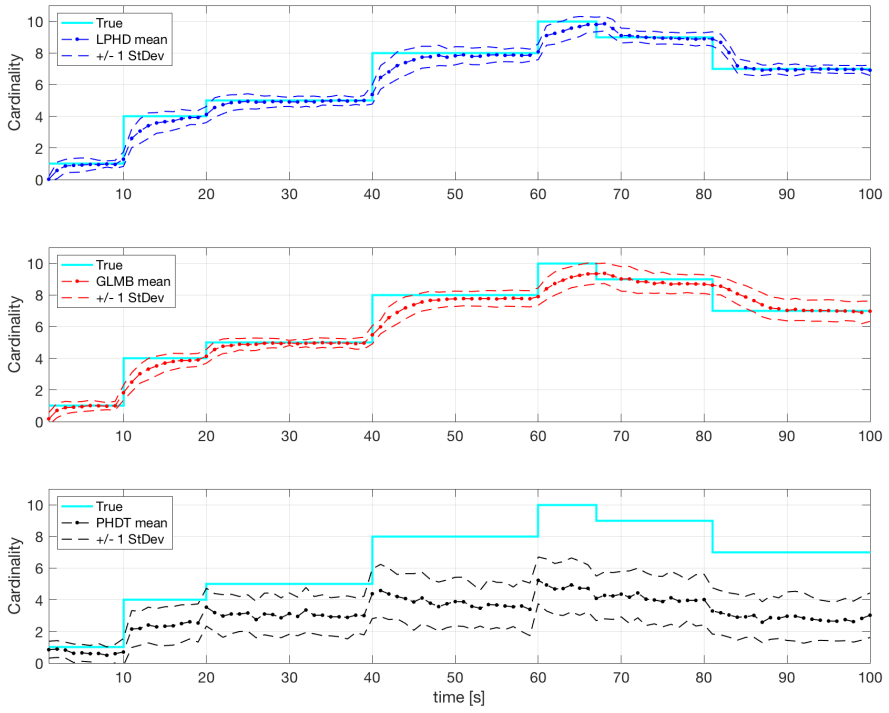

Figure 2.5: Cardinality statistics for the LPHD, δ-GLMB, and PHDT filters with clutter rate $\lambda_c = 10$ and $p_D = 0.75$, averaged over 100 Monte Carlo runs.

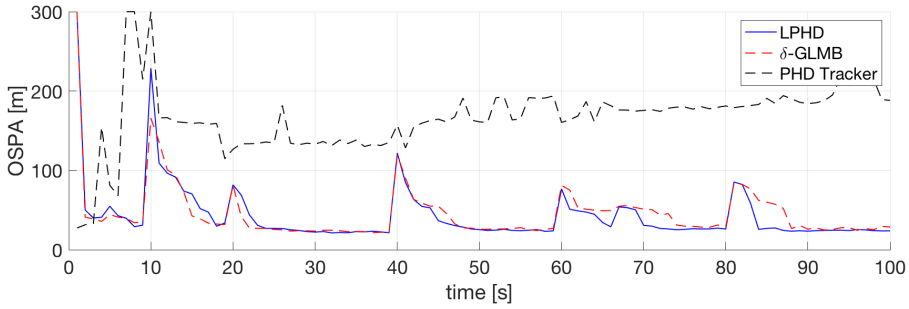

Figure 2.6: OSPA distance ($p = 1, c = 300m$) for LPHD, δ-GLMB, and PHDT filters with clutter rate $\lambda_c = 10$ and $p_D = 0.75$, averaged over 100 Monte Carlo runs.

2.7 Conclusion

The theory of Labeled Random Finite Set (LRFS) provides a theoretical foundation for Bayes-optimal multi-target tracking. A well-known method based on this approach is the Generalized Labeled Multi-Bernoulli (GLMB) filter, the first computationally tractable and provably Bayes-optimal multi-target tracker (see Section 1.3.6). This chapter explored the following questions: if an analogous labeled version of the PHD filter exists, what is its relationship with the GLMB filter? Is it a suboptimal but computationally faster alternative? This chapter proved that these questions can be answered affirmatively, and the Labeled Multi-Bernoulli (LMB) filter proposed by Reuter et al. can be reinterpreted as a Labeled Probability Hypothesis Density (LPHD) filter.

In contrast to the heuristic solutions for the PHD filter with labels discussed in Section 2.2, the proposed LPHD filter completely implements the labeled RFS theory, which facilitates the representation of labeled objects and presents a more accurate approximation of the multi-object Bayes filter. However, the computational complexity of the LPHD filter update is significantly higher than other implementations of the PHD filter using the standard RFS. On the other hand, the LPHD filter can be considered an approximation of the δ-GLMB filter, which approximates the multi-object posterior using an LMB RFS in order to simplify the prediction step. This approximation results in an accurate and real-time target tracking algorithm. Additionally, the LPHD filter facilitates an implementation using Gaussian mixtures and does not tend to degrade when the clutter is dense or the objects are close.

The LPHD filter provides several interesting topics for future research: due to the LMB RFS properties, it is possible to calculate the variance of the number of targets in a simple manner, using the idea proposed by Schlangen et al. [Sch+18] to create a second-order LPHD, which can aid integration with sensor management platforms. Another topic requiring further exploration are the truncation techniques that can be implemented to reduce filter complexity. Further possible extensions of the LPHD filter include the Sequential Monte Carlo (SMC) implementation, the support of multi-model approach, the application to Simultaneous Localization and Mapping (SLAM), and the development of a different birth model.

Chapter 3

The multi-sensor version of the LPHD filter

The mono-sensor Labeled Probability Hypothesis Density (LPHD) filter can only be used with the standard multi-object measurement model, which presumes that an object generates at most a single measurement (a detection), that any measurement is generated by at most a single object, and that measurements not generated by objects are false detections or clutter. In contrast, sensors such as radars and sonars do not obey the standard measurement model because their measurements are a sum of the signals collected by all objects in the scene: they are superpositional sensors.

Multi-sensor systems and superpositional sensors are everywhere, and multi-object tracking filters specifically designed for them could offer a significant improvement over conventional, mono-sensor standard-detection tracking approaches. In 2009, Mahler presented the theoretical derivation of the unlabeled multi-sensor PHD filter for the two-sensor case [Mah09b; Mah09c]. Later, Delande introduced a rigorous development of the unlabeled PHD filter for the multi-sensor case in 2010 [Del+10; Del12], and Liu proposed another unlabeled multi-sensor extension limited to linear sensor systems in 2011 [LW11].

This chapter presents the rigorous development of the multi-sensor LPHD filter as an extension of the mono-sensor LPHD filter for superpositional sensors based on the work of Papi, Saucan, and Mahler et al. [PK15; SLC17; Mah18]. The proposed multi-sensor labeled PHD filter differs from the filters mentioned above in that its development is based on the Labeled Random Finite Set (LRFS) formalism and the Labeled Multi-Bernoulli (LMB) RFS distribution, rather than using the standard RFS formalist and Poisson RFS distribution.

3.1 Superpositional measurement model

The Labeled Random Finite Set (LRFS) theory is the first rigorous theoretical formulation of true multi-object tracking, and is the basis for the first implementable and provably Bayes-optimal multi-object tracking algorithm known as the Generalized Labeled Multi-Bernoulli (GLMB) filter [Mah14]. The GLMB filter is an exact closed-form solution to the multi-object Bayes filter with respect to the standard multi-object measurement model and the separable model for pixelized image data [Pap+13; PPH14; Mah14; Mah17; Mah18]. A standard measurement model can be expressed by:

$$Z_k = \eta_k(x) + V_k \tag{3.1}$$

where x is the state of a single object, $\eta_k(x)$ is a nonlinear measurement function, and V_k is a zero-mean random noise vector. The measurement density (sensor likelihood function) for this model is defined by:

$$L_z(x) = g_k(z|x) = g_{V_k}\Big(z - \eta_k(x)\Big) \tag{3.2}$$

For sensors like radars and sonars that do not obey the standard measurement model because they are superpositional, the measurement model is given by:

$$Z_k = \eta_k(X) + V_k = \eta_k(x_1) + \cdots + \eta_k(x_n) + V_k \tag{3.3}$$

where $X = \{x_1, \cdots, x_n\}$ with $|X| = n$ is the multi-object state set. In general, $\eta_k(x)$ and V_k are complex-valued vectors. The measurement density of this model is:

$$\begin{aligned} L_z(X) &= g_k(z|X) \\ &= g_{V_k}(z - \eta_k(X)) \\ &= g_{V_k}(z - \eta_k(x_1)) - \cdots - g_{V_k}(z - \eta_k(x_n)) \end{aligned} \tag{3.4}$$

The multi-object Bayes filter for superpositional models is computationally intractable. To address this issue, Mahler proposed a theoretical formulation for the general superpositional Cardinalized PHD (CPHD) filter in 2009 [Mah09a]. In 2012 and 2013, Mahler and Nannuru et al. introduced computationally tractable unlabeled superpositional PHD and CPHD filters [ME12; NCM13]. During 2015 and 2017, Papi and Saucan et al. proposed some extension to the GLMB filter to make it capable of handling labeled superpositional measurements models [PK15; SLC17]. Both approaches are computationally demanding for a large numbers of objects.

To model the superpositional tracking filters within the LRFS, Papi and Kim generalized the GLMB filter's measurement-update step to arbitrary multi-object measurement models [PK15]. They explored the fact that the measurement-updated multi-object distribution is generally not a GLMB, and any labeled multi-object distribution can be approximated by a GLMB distribution that has the same PHD and cardinality distribution [Pap+14]. However, their superpositional GLMB filter is computationally tractable only for a few number of objects. In 2017, Saucan et al. proposed a modified approach to the Papi and Kim filter to make it less computationally expensive [SLC17]. However, their approach is still computationally demanding for larger number of targets.

3.2 The multi-sensor LPHD filter

This section presents the development of the multi-sensor LPHD filter, which is based on the superpositional measurement model. Throughout the sections, we use the notations and abbreviations presented in Sections 1.2.1, 1.3.2, and Appendix A. We also use the Probability Generating Functional (PGFl) to prove the propositions necessary for the derivation of the proposed filter (see Section 2.1). The following presents complementary information based on Mahler's work [Mah14; Mah17; Mah18].

A Labeled Multi-Bernoulli (LMB) distribution can be represented by:

$$\pi(\boldsymbol{X}) = \Delta(\boldsymbol{X})w^J(\mathcal{L}(\boldsymbol{X}))p^{\boldsymbol{X}}, \qquad w(L)^J = \prod_{\ell\in J-L}\left(1-r^{(\ell)}\right)\prod_{\ell\in L}\frac{1_L(\ell)r^{(\ell)}}{1-r^{(\ell)}} \tag{3.5}$$

for some $J \subseteq \mathbb{L}$ with $0 \leq r^{(\ell)} \leq 1$ for all $\ell \in J$. Its PGFl is defined by:

$$G[h] = \prod_{\ell\in\mathbb{L}}(1 - r^{(\ell)} + r^{(\ell)}p^{(\ell)}[h]) \tag{3.6}$$

Similar to a Poisson RFS characterized by its PHD, an LMB RFS Ξ is also characterized by its labeled PHD [Mah17]:

$$\nu_\Xi(x,\ell) = \left[\frac{\delta G_{k|k}}{\delta(x',\ell')}[h]\right]_{h=1} = 1_J(\ell)r^{(\ell)}p^{(\ell)}(x) = 1_J(\ell)r^{(\ell)}p(x,\ell) \tag{3.7}$$

however, an LMB RFS presents additional advantages over a Poisson RFS employed in the conventional unlabeled PHD filter. The cardinality distribution of an LMB RFS can be calculated by [Mah18]:

$$G_\Xi(x) = \prod_{\ell\in J}(1 - r^{(\ell)} + r^{(\ell)}x) \tag{3.8}$$

consequently, the mean and variance of its cardinality distribution are easily calculated by:

$$\mu_\Xi = \sum_{\ell\in J} r^{(\ell)}, \qquad \sigma_\Xi^2 = \sum_{\ell\in J} r^{(\ell)}(1-r^{(\ell)}) \tag{3.9}$$

where $\sigma_\Xi^2 \leq \mu_\Xi$, while the more restrictive case $\sigma_\Xi^2 = \mu_\Xi$ holds a Poisson RFS.

3.2.1 Multi-sensor LPHD prediction

The prediction equation for the multi-sensor LPHD filter is the same as that for the mono-sensor LPHD filter, and is given by:

$$\nu_{k|k-1}(x,\ell) = \begin{cases} \int p_S(x',\ell)\, f_{k|k-1}(x|x')\, \nu(x',\ell)\mathrm{d}x & \text{if } \ell \in \mathbb{L} \\ \nu^B(x,\ell) & \text{if } \ell \in \mathbb{B} \end{cases} \tag{3.10}$$

where $\mathbb{B} \subseteq \mathbb{L}_{k|k-1}$ is the labeled set for the birth targets and $\mathbb{L}$ is the labeled set for the current targets (see Section 2.4.1 for further details).

3.2.2 Multi-sensor LPHD update

For a labeled RFS $\boldsymbol{X} = \{(x_1, \ell_1), \cdots, (x_n, \ell_n)\}$ with $|\boldsymbol{X}| = n$ and $(\ell_1, \cdots, \ell_n)$ distinct, the superpositional measurement model has the following form:

$$Z_k = \eta_k(\boldsymbol{X}) + V_k = \eta_k(x_1, \ell_1) + \cdots + \eta_k(x_n, \ell_n) + V_k \tag{3.11}$$

with the measurement distribution defined by:

$$L_z(\boldsymbol{X}) = g_k(z|\boldsymbol{X}) = g_{V_k}(z - \eta_k(\boldsymbol{X})) \tag{3.12}$$

Assume that at any time k, the measurement noise for the superpositional measurement model is approximately Gaussian and expressed by $g_{V_k}(z) \approx \mathcal{N}(\cdot; z, R_k)$. Also consider that the distribution of $Z_k = \eta_k(\Xi_{k|k-1})$ is Gaussian.

Proposition 4. *Then the multi-sensor LPHD update for a predicted LPHD $\nu_{k|k-1}(x, \ell)$ and for new superpositional measurements Z_k is defined by:*

$$\nu_k(x, \ell) = \begin{cases} 0 & \text{if } \ell \notin J \\ \dfrac{\alpha_k \, \nu_{k|k-1}(x, \ell) \; \mathcal{N}(z; z_k - \eta(x, \ell) - o^{(\ell)}, R + O^{(\ell)})}{\max_{\ell' \in J} \int \nu_{k|k-1}(x', \ell') \; \mathcal{N}(z; z_k - \eta(x', \ell') - o^{(\ell')}, R + O^{(\ell')}) \, \mathrm{d}x'} & \text{otherwise} \end{cases} \tag{3.13}$$

where $0 < \alpha_k \leq 1$ is an application-dependent constant, and $o^{(\ell)}$ and $O^{(\ell)}$ are defined by:

$$o^{(\ell)} = \sum_{\ell' \in J - \{\ell\}} \nu^{(\ell')}[\eta], \tag{3.14}$$

$$O^{(\ell)} = \sum_{\ell' \in J - \{\ell\}} \left(\nu_{(\ell')}[\eta\eta^T] - \nu^{(\ell')}[\eta] \nu^{(\ell')}[\eta]^T \right) \tag{3.15}$$

where ^{T} denotes the matrix transpose, and:

$$\nu^{(\ell)}[\eta] = \int \eta(x, \ell) \nu_{k|k-1}(x, \ell) \mathrm{d}x, \tag{3.16}$$

$$\nu^{(\ell)}[\eta\eta^T] = \int \eta(x, \ell) \eta(x, \ell)^T \nu_{k|k-1}(x, \ell) \mathrm{d}x, \tag{3.17}$$

and similar to the mono-sensor LPHD filter, $\int \nu_{k|k-1}(x, \ell) \mathrm{d}x \leq 1$ for every ℓ.

Proof. The approach is similar to that used by Mahler to derive the unlabeled superpositional PHD and CPHD [ME12; Mah14; Mah17; Mah18]. Considering that the measurement-updated multi-object probability distribution is given by Bayes' rule:

$$\pi_k(X_k|Z_{1:k}) = \frac{g_k(Z_k|X_k)\pi_{k|k-1}(\boldsymbol{X}_k|Z_{1:k-1})}{g_k(Z_k|Z_{1:k-1})} \tag{3.18}$$

$$g_k(Z_k|Z_{1:k-1}) = \int g_k(Z_k|X_k)\pi_{k|k-1}(\boldsymbol{X}_k|Z_{1:k-1})\delta X_k \tag{3.19}$$

then the measurement-updated PHD is:

$$\begin{aligned}\nu_k(x,\ell) &= \int \pi_k(\{(x,\ell)\}) \cup W)\delta W \\ &= \frac{\int g_{V_k}(z_k - \eta(x,\ell) - \eta(W))\pi_{k|k-1}(\{(x,\ell)\}) \cup W)\delta W}{g_k(Z_k|Z_{1:k-1})}\end{aligned} \tag{3.20}$$

The derivation is realized in two steps following Nannuru and Mahler's approximations [NCM13; Mah14; Mah18].

Step 1: Approximate the Bayes normalization factor. This is possible via the change of variables for the set integral (see Appendix A):

$$\begin{aligned}g_k(Z_k|Z_{1:k-1}) &= \int \mathcal{N}(z; z_k - \eta(\boldsymbol{X}), R)\pi_{k|k-1}(\boldsymbol{X}_k|Z_{1:k-1})\delta\boldsymbol{X} \\ &= \int \mathcal{N}(z; z_k - z, R)P(z)\mathrm{d}z\end{aligned} \tag{3.21}$$

where $P(z)$ represents the distribution of the random vector $Z = \eta(\Xi)$. Assuming that $\pi_{k|k-1}(\boldsymbol{X}_k|Z_{1:k-1})$ is an LMB RFS with LPHD and PGFl:

$$\begin{aligned}\nu(x,\ell) = \nu^{(\ell)}(x) &= 1_J(\ell)r^{(\ell)}p(x,\ell), \\ G[h] &= \prod_{\ell\in\mathbb{L}}(1 - r^{(\ell)} + r^{(\ell)}p^{(\ell)}[h])\end{aligned}$$

define o and O as the expected value and covariance matrix of $P(z)$, and approximate it with $P(z) \approx \mathcal{N}(z; z - o, O)$. Then the Bayes normalization factor is given by:

$$g_k(Z_k|Z_{1:k-1}) \approx \mathcal{N}(z; z_k - o, R + O) \tag{3.22}$$

then writing o and O for the labeled case:

$$o = \sum_{\ell} \int \eta(x,\ell)\nu(x,\ell)\mathrm{d}x \tag{3.23}$$

$$\begin{aligned} O = &\sum_{\ell} \int \eta(x,\ell)\eta(x,\ell)^T \nu(x,\ell)\mathrm{d}x \\ &+ \sum_{\ell_1,\ell_2} \int \eta(x_1,\ell_1)\eta(x_2,\ell_2)^T \Big[\nu_{2,k|k-1}(x_1,\ell_1,x_2,\ell_2) \\ &- \nu_{k|k-1}(x_1,\ell_1)\nu_{k|k-1}(x_2,\ell_2)\Big]\mathrm{d}x_1\mathrm{d}x_2 \end{aligned} \tag{3.24}$$

and finally it can be shown that:

$$o = \sum_{\ell} \nu^{(\ell)}[\eta], \qquad O = \sum_{\ell} \left(\nu^{(\ell)}[\eta\eta^T] - \nu^{(\ell)}[\eta]\nu^{(\ell)}[\eta]^T\right)$$

Step 2: For a determined (x,ℓ), define the new labeled multi-object distribution and calculate the approximate PHD:

$$\pi^{(x,\ell)}_{k|k-1}(\boldsymbol{X}_k|Z_{1:k-1}) = \nu(x,\ell)^{-1}\pi(\{(x,\ell)\} \cup \boldsymbol{X}) \tag{3.25}$$

and using the change of variables formula for the set integral (see Appendix A):

$$\frac{\nu_k(x,\ell)}{\nu(x,\ell)} = \frac{\int \mathcal{N}(z; z_k - \eta(\boldsymbol{X}) - \eta(\boldsymbol{X}), R)\pi^{(x,\ell)}_{k|k-1}(\boldsymbol{X}_k|Z_{1:k-1})\delta\boldsymbol{X}}{g_k(Z_k|Z_{1:k-1})} \tag{3.26}$$

$$= \frac{\int \mathcal{N}(z; z_k - \eta(\boldsymbol{X}) - z, R)P^{(x,\ell)}(z)\mathrm{d}z}{g_k(Z_k|Z_{1:k-1})} \tag{3.27}$$

where $P^{(x,\ell)}(z)$ is the distribution of the random vector $Z^{(x,\ell)} = \eta(\Xi^{(x,\ell)})$ while $\pi^{(x,\ell)}_{k|k-1}(\cdot)$ is the distribution of the LRFS $\Xi^{(x,\ell)}$. Define $o^{(x,\ell)}$ and $O^{(x,\ell)}$ as the expected value and covariance matrix of $P^{(x,\ell)}(z)$. These values become independent of x and we can express $o^{(\ell)} =^{(x,\ell)}$, $O^{(\ell)} = O^{(x,\ell)}$, and by approximating $P^{(x,\ell)}(z) \approx \mathcal{N}(z; z - o^{(\ell)}, O^{(\ell)})$, we have:

$$\nu_k(x,\ell) \approx \frac{\nu_{k|k-1}(x,\ell)\ \mathcal{N}(z; z_k - \eta(x,\ell) - o^{(\ell)}, R + O^{(\ell)})}{\mathcal{N}(z; z_k - o^{(\ell')}, R + O^{(\ell')})} \tag{3.28}$$

then integrate both sides of the above equation:

$$\int \nu_k(x,\ell)\mathrm{d}x \approx \frac{\int \nu_{k|k-1}(x,\ell)\ \mathcal{N}(z; z_k - \eta(x,\ell) - o^{(\ell)}, R + O^{(\ell)})\mathrm{d}x}{\mathcal{N}(z; z_k - o^{(\ell')}, R + O^{(\ell')})} \tag{3.29}$$

the left side must be no larger than 1; however, the same is not necessarily true of the right side because it is an approximation. According to Mahler [Mah18], to ensure that the right side of the above equation never exceeds 1, it is necessary to utilize heuristic reasoning, such as normalizing the PHD as follows:

$$\nu_k(x,\ell) \approx \frac{\nu_{k|k-1}(x,\ell)\ \mathcal{N}(z; z_k - \eta(x,\ell) - o^{(\ell)}, R + O^{(\ell)})}{\max_{\ell' \in J} \int \nu_{k|k-1}(x',\ell')\ \mathcal{N}(z; z_k - \eta(x',\ell') - o^{(\ell')}, R + O^{(\ell')})\,\mathrm{d}x'} \tag{3.30}$$

where $\int \nu_{k|k-1}(x', \ell')\mathrm{d}x' \leq 1$ for all $\ell \in J$. However, this would mean that there is always at least one track with a unity probability of existence where $\int \nu_{k|k-1}(x', \ell')\mathrm{d}x' = 1$, which is physically improbable. This can be fixed using an heuristic factor $0 < \alpha_k \leq 1$ as follows:

$$\nu_k(x, \ell) = \frac{\alpha_k \, \nu_{k|k-1}(x, \ell) \; \mathcal{N}(z; z_k - \eta(x, \ell) - o^{(\ell)}, R + O^{(\ell)})}{\max_{\ell' \in J} \int \nu_{k|k-1}(x', \ell') \; \mathcal{N}(z; z_k - \eta(x', \ell') - o^{(\ell')}, R + O^{(\ell')}) \, \mathrm{d}x'} \tag{3.31}$$

■

3.3 Conclusion

Most of the multi-object tracking algorithms in the current literature are developed for single sensors. The use of multiple sensors can reduce uncertainty about object states, the probability of object existence, and the variance in the cardinality estimation. However, this problem is generally computationally intractable, especially for more than two sensors.

The multi-object Bayes-optimal filter for non-standard measurement models is still computationally intractable. An attempt to address this issue started with Mahler in 2009, who proposed a theoretical formulation of the general superpositional CPHD filter, followed by other publications with Nannuru et al. in 2012 and 2013. During 2015 and 2017, an extension of the GLMB filter was proposed by Papi and Saucan et al. to make it capable of handling labeled superpositional measurements models. But both Papi and Saucan's approaches are computationally demanding for a large number of objects.

This chapter can be seen as a continuation of the aformentioned works to derive a tractable multi-sensor multi-object Bayes filter within the labeled RFS. It was presented the rigorous development of the multi-sensor LPHD filter for superpositional sensors. The proposed derivation is also applicable to approximations such as the Labeled Multi-Bernoulli (LMB) filter since this filter requires a special case of the GLMB RFS [Reu+14]. Possible implementations of the multi-sensor LPHD filter can be based on the work of Papi and Beard et al. [Pap+14; BVV15].

Chapter 4

Sensor Management for multi-object tracking

Advances in sensor technologies led to the emergence of a large number of controllable degrees of freedom in sensing devices, which allow software command update parameters such as center frequency, bandwidth, beam-pointing, and many other aspects of sensors' resources and operating modes. Sensor systems operate under resource constraints that prevent the constant simultaneous use of all resources; sensor management becomes relevant when the sensing system has the capability of actively managing these resources in reaction to previous measurements [Her+08; HC11]. Mutli-object sensor management is typically an optimal non-linear control problem. Its objective is to allocate resources optimally by directing the right sensor on the proper platform to the appropriate target at the correct time [Mah03b; Her+08]. In order to accomplish this objective, mutli-object sensor management must address issues such as the number of targets varying in time and the measurements' susceptibility to missed detections and false alarms.

With reference to the Random Finite Set (RFS) approach, Mahler developed theoretical foundations of the mutli-object sensor management reward function related to the Posterior Expected Number of Targets (PENT) [MZ04], and its variation to prioritize targets of interest, the Posterior Expected Number of Targets of Interest (PENTI) [Mah04; Mah14]. Delande, in turn, introduced a reward function called Balanced Explorer and Tracker (BET) that provides efficient sensor management in situations where the sensor's field of view (FOV) cannot cover the whole state space at the same time [Del12]. Ristic and Vo proposed a reward function to sensor management using the Rényi divergence between the mutli-object prior and posterior densities [RV10; RVC11]. More recently, Hoang, Vo, Vo, and Mahler offered a new intuitive and tractable objective function based on the Cauchy-Schwarz information functional [Hoa+15].

When only a subset of the total targets can be successfully tracked, the prioritization of target tracks is crucial and cannot be achieved by means of information gain-based metrics. To overcome the limitations of existing metrics, recent research started to apply a statistical risk model used to calculate an expected cost as a metric. Papageorgiou et al. proposed a Risk-based approach to sensor resource management for the problem of missile defense [PR07]. Wang et al. developed a Bayesian Risk-based sensor management for integrated detection and estimation [WHE11]. Martin introduced a statistical Risk-based metric for a field of view problem [Mar15]. This thesis differs from the above-mentioned studies presenting a Risk-based sensor management using the RFS and the Partially Observable Markov Decision Process (POMDP) framework.

This chapter is devoted to sensor management using the Extended Kalman filter (EKF) and the Probability Hypothesis Density (PHD) filter and provides a clarified version of the author's conference papers [Gom+16; Gom+17]. Section 4.1 presents the existing sensor management approaches, such as Task-based, information-theoretic, and mission-oriented. This section also discusses in detail the existing sensor management schemes that combine the RFS theory and the POMDP framework. Section 4.2 introduces the newly derived analytical expression of the Expected Risk Reduction (ERR) metric between two densities. We define the Expected Risk Reduction (ERR) approach to mutli-object sensor management as similar to that in [PR07; WHE11; Mar15; Gom+16]; however, to jointly estimate the number of targets and their states, the Gaussian-Mixture PHD (GM-PHD) tracker is implemented. Finally, Sections 4.3 and 4.4 demonstrate the performance of the proposed reward function in the context of multi-object tracking in a radar beam-pointing problem where targets need to be prioritized.

4.1 Fundamentals of Sensor Management

The term sensor management refers to the control that an agile system has over the sensor configuration parameters to satisfy operational constraints and achieve operational objectives. To accomplish this, one typically seeks a policy for determining the optimal sensor configuration at each time, within constraints, as a function of information available from *a priori* measurements and other sources [BP99; XS02; Her+08; HC11; BWT11]. For a given sensing application, sensor management consists of determining sensing actions that maximize the efficiency of the resulting sensor measurements. Depending on the complexity of the system and the number of sensing actions available, optimal sensor management can be intractable. For most applications of interest, a large number of decisions must be made regarding how sensors should collect measurements, making sensor management challenging [BP99; XS02; Her+08; HC11; BWT11].

One of the earliest and most challenging applications of sensor management is known as multi-object tracking, which refers to the problem of jointly estimating the number of targets and their states or trajectories from noisy sensor measurements [BP99; BWT11; Mah14]. Sensor management in multi-object tracking is typically an optimal nonlinear control problem where the number of targets varies randomly in time and the measurements are susceptible to missed detections and false alarms. Thus, the goal is to allocate resources optimally, directing the right sensor on the proper platform to the appropriate target at the correct time [Mah03b; Her+08].

In many multi-object tracking applications, a sensor can be controlled by changing the position, orientation, or motion of the sensor platform or by operating it in a different mode, which may have a significant impact on the quality of the estimation performance of the target tracking system. Modern surveillance systems often employ multiple controllable sensors capable of collecting information on objects of interest in their field of view. These sensors must coordinate their observation strategies to enhance the information that will be collected by their future measurements in order to estimate the states of objects of interest [BP99; XS02; Her+08; HC11].

Beginning in the 1990s, the rapidly growing interest in sensor management was associated in large part with developments in sensor and communications technologies. Recently, new sensor management approaches based on modeling sensor management as a decision process have emerged that provide a unifying perspective for current state-of-the-art sensor manage-

ment research. A decision process is a time sequence of measurements and control actions in which each action in the sequence is followed by a measurement acquired as a result of the previous action. A sensor manager is designed to allow for the specification of a decision rule, often called a policy, that generates realizations of the decision process. An optimal policy will generate decision processes that, on average, will maximize an expected reward [Her+08; HC11; BWT11; Mah14].

A natural way to simplify the task of policy optimization is to assume that the general decision process satisfies some additional Markovian properties. To make the general decision process Markovian, the assumption is that the state sequence is dependent only on the most recent state and the action given only on the entire past. When the state can be recovered from the measurements, the resultant process is called a Markov Decision Process (MDP). When the state is not recoverable, the resultant process is called a Partially Observable Markov Decision Process (POMDP) [Her+08; HC11].

4.1.1 Partially Observed Markov Decision Problems

Sensor management represents sequential decision-making systems where each decision generates new observations that provide additional information. The decisions are made in the presence of uncertainty, both in the state and the observation, and are based only on the past measurement. This class of problems has been studied in the Partially Observable Markov Decision Process (POMDP) framework.

In this framework, the multi-object dynamics are modeled as a Markov process, but only the posterior probability density function of the multi-object state is known, and the true underlying state is unknown. The measurements follow a known distribution, which is conditional on the multi-object state and the sensor control action. The benefit of performing a given action is expressed by a reward function, which characterizes the objectives of the control system. Every time a decision is needed, the goal is to find the control action that maximizes this reward function.

Sensor management as a decision process assumes that a sensor collects a data sample z_k at time k after taking a sensing action u_{k-1} at time $k-1$. It is typically assumed that the possible actions are selected from a finite action space $\mathbb{U}_k$, which may change over time. The selected action u_k depends only on past samples $\{z_k, z_{k-1}, \cdots, z_1\}$ and past actions $\{u_{k-1}, u_{k-2}, \cdots, u_0\}$, where the initial action u_0 is determined offline. The function that maps previous data samples and actions to current actions is called a policy. That is, at any time k, a policy specifies a mapping γ_k and, for a specific set of samples, an action $u_k = \gamma_k(\{u_k\}, \{z_{k+1}\})$. A decision process is a sequence $\{u_k, z_{k+1}\}_{k\geq 0} = \{u_0, z_1, u_1, z_2, u_2, z_3, \cdots, u_k, z_{k+1}\}$, which is typically random and can be viewed as a realization from a generative model specified by the policy and the sensor measurement statistics [Her+08; HC11].

The elements of a POMDP include object state at time k represented by the predicted multi-object posterior pdf $\pi_{k|k-1}(X_k|Z_{1:k-1})$, a set of admissible sensor actions $\mathbb{U}_k$, and a reward function associated with different control selection $\mathcal{R}(\cdot)$. The problem lies in the fact that at the time we want to carry out a control action, we have no knowledge of the posterior density that would arise from taking that action. The optimal control action is given by maximizing the expected value of a reward function over the set of admissible

actions illustrated in Figure 4.1 is defined accordingly [Her+08]:

$$u_k = \arg\max_{u \in \mathbb{U}_k} \mathbb{E}\left[\mathcal{R}\Big(u, \pi_{k|k-1}(X_k|Z_{1:k-1}), Z_k(u)\Big)\right] \tag{4.1}$$

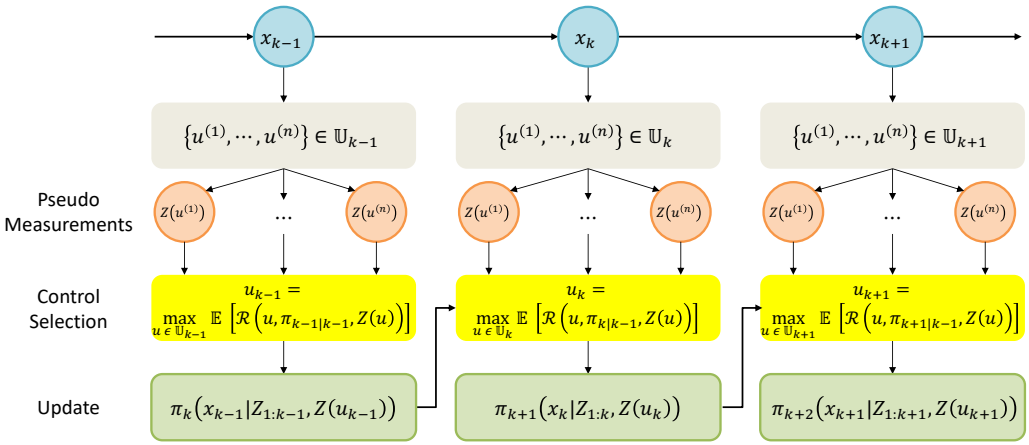

Figure 4.1: The Partially Observable Markov Decision Process framework for Sensor Management and multi-object Tracking. The term u_k denote the best control to be applied at time k; $\mathcal{R}\Big(u, \pi_{k|k-1}(X_k|Z_{1:k-1}), Z_k(u)\Big)$ is the real-valued reward function associated with the control u; $Z_k(u)$ is a predicted measurement obtained if the control u is applied, and can be obtained using the Predicted Ideal Measurement Set (PIMS) as proposed by Mahler [Mah04; Mah14]. The function $\mathcal{R}(\cdot)$ is generally based either on a decrease of uncertainty or increase of information gain between the predicted and posterior multi-object densities. The following presents an overview of the existing sensor management approaches and how they calculate the reward function.

4.1.2 Task-based Sensor Management

In order to produce Bayes-optimal sensor management results, one proposed suggestion is to optimize quantities that are relevant to the sensing tasks and the operational goal of a system, hence the name Task-based sensor management. The objective function of this approach is formulated as a cost function which usually depends on the performance metrics such as error or cardinality variance. For example, when an area surveillance is performed using radar measurements, one could a) maximize the expected probability of detecting a target, b) maximize the expected signal-to-noise ratio of a measurement, or c) minimize the expected uncertainty in the position estimate of the target [BP99; XS02; KHK05; Her+08].

Task-based schemes result in Bayes-optimal solutions. However, their main disadvantage is that, most often, they optimize quantities that are relevant to the operational goal but are not explicitly what the user actually needs. Consider, for example, an air-traffic-control scenario: minimizing the uncertainty in position and velocity of all aircrafts is beneficial but what an operator is truly interested in is whether two aircrafts are going to collide.

One of the most common approaches when tracking a target is to select the sensing action that will optimize a covariance-based measure. The trace of the covariance matrix is usually considered when tracking a target using a Kalman filter and a sensing action is selected so that its expected value is minimized. The trace is preferred over other matrix functions because it is simple to evaluate and intuitive to explain. The uncertainty in the estimated density is

managed, assuming that it can be described sufficiently well by a covariance matrix. Under the Task-based sensor management approach, one can also obtain myopic and non-myopic solutions given certain assumptions that allow the problem to be formulated as a Partially Observable Markov Decision Process (POMDP) [Her+08; Mah14].

In multi-object tracking, when the true number of targets is not known, the trace of the covariance matrix, or any of the other commonly used covariance-based schemes, cannot be used as measure of uncertainty because the state space is not Euclidean [Mah14]. This is due to the existence variable of each track taking values in [0,1] instead of the reward function $\mathcal{R}(\cdot)$. To overcome this problem and to take into account the limited field of view of a sensor, Mahler developed theoretical foundations of the multi-object sensor management reward function within the Random Finite Set (RFS).

Posterior Expected Number of Targets

The Posterior Expected Number of Targets (PENT) sensor management was developed by Mahler to solve the problem of the optimal placement of the sensor's field of view (FOV) [MZ04]. PENT is an objective function developed using the Predicted Ideal Measurement Set (PIMS), and it selects the control action that maximizes the number of targets seen by the sensor [Mah14]. By maximizing the value of the PENT reward function, the sensor's FOV can be directed to those places where the sensor is most likely to collect the PIMS. When a Probability Hypothesis Density (PHD) filter is used, the PENT reward function is given as follows:

$$u_k = \arg\max_{u \in \mathbb{U}_k} \left[\int \nu_{k|k}\left(x | Z_k^{PIMS}(u)\right) \right] \tag{4.2}$$

where u_k is the best sensor control to be applied at time k, and $Z_k^{PIMS}(u)$ is a simulated ideal set of measurements based on PIMS, which will be obtained if the control u is applied, as proposed by Mahler [Mah04; Mah14].

Given a selected sensor command, the PIMS is comprised of clutter-free and noise-free measurements that are most likely to be obtained from the selected sensor. Considering $N_{k|k-1}$ the predicted number of objects, $X_{k|k-1} = \{x_{k|k-1}^{(i)}\}_i^{N_{k|k-1}}$ the predicted multi-object state, $u \in \mathbb{U}_k$ the sensor control, and $g(z|x,u)$ a measurement likelihood function that is dependent on the control u applied to the sensor, then the PIMS can be obtained as follows [Mah04; Mah14]:

$$Z_k^{PIMS}(u) = \bigcup_{i=1}^{N_{k|k-1}} \left\{ \arg\max_z g(z|x_{k|k-1}^{(i)}, u) \right\} \tag{4.3}$$

Posterior Expected Number of Targets of Interest

The PENT sensor management was also extended to consider the tactical significance of a target, resulting in the Posterior Expected Number of Targets of Interest (PENTI) sensor

management. The PENTI reward function for the PHD filter is given by [Mah04; Mah14]:

$$u_k = \arg\max_{u \in \mathbb{U}_k} \left[\int \tau(x)\ \nu_{k|k}\Big(x|Z_k^{PIMS}(u)\Big) \right] \tag{4.4}$$

where $\tau(x) \in [0, 1]$ is the tactical importance function of each target.

4.1.3 Information-Theoretic Sensor Management

The information-theoretic sensor management aims to improve the information content of the multi-object distribution by optimizing some measure of information gain. It has attracted substantial research interest due to its performance for multi-object tracking [Kas97; Her+08; HC11; Mah14].

The idea of measuring information gain as a reward function in the sensor control problem first appeared in 1991 with the works of Hintz and McVey [Hin91; HM91]. In 1992, Manyika and Durrant-Whyte proposed the expected value of the information gain to address the sensor management problem [MD92]. Information gain functions are theoretically rigorous; however, they express no discernible relationship with operational requirements and in general do not have an intuitive physical interpretation. The following presents the most important information measures used for multi-object tracking.

Rényi divergence

Rényi or alpha divergence measures information gain between two probability densities. The Rényi divergence between any two densities, $p_0(x)$ and $p_1(x)$, is defined by [KKH03; Her+08; vH14; Mah14]:

$$D_\alpha(p_0, p_1) = \frac{1}{\alpha - 1} \log \int \Big[p_1(x)\Big]^\alpha \Big[p_0(x)\Big]^{1-\alpha} \mathrm{d}x \tag{4.5}$$

where the parameter $\alpha \geq 0$ determines how much the tails of the two distributions are emphasized. In the special case where $\alpha \to 1$, the Rényi divergence becomes the Kullback-Leibler divergence [Her+08; vH14]. Kreucher, Ristic and Hoang employed the Rényi divergence as the generalization of the Kullback-Leibler divergence in order to solve the multi-object sensor management problem [KKH03; KHK05; RV10; RVC11; HV14]. They used the Rényi divergence to calculate the reward function between the multi-object posterior and predicted densities as shown below:

$$D_\alpha(\pi_{k|k}, \pi_{k|k-1}) = \frac{1}{\alpha - 1} \log \int \Big[\pi_{k|k}(\cdot)\Big]^\alpha \Big[\pi_{k|k-1}(\cdot)\Big]^{1-\alpha} \mathrm{d}x \tag{4.6}$$

where:

$$\pi_{k|k}(\cdot) = \pi_{k|k}(x_k|Z_{1:k-1}, u_{0:k-1}, Z_k(u_k))$$
$$\pi_{k|k-1}(\cdot) = \pi_{k|k-1}(x_k|Z_{1:k-1}, u_{0:k-1})$$

The divergence $D_\alpha(\cdot)$ can be used to replace $\mathcal{R}(\cdot)$ in (4.1). The multi-object prior density $\pi_{k|k-1}(\cdot)$ represents the target state obtained until the last sensor control $u_{0:k-1}$, while the multi-object posterior density $\pi_{k|k}(\cdot)$ is the target state that will be obtained if the sensor control action u_k is performed. The objective is to maximize the information gain by computing the Rényi divergence between the predicted and updated distribution choosing the right sensor control u_k. Any choice of the sensor control u_k can lead to different measurement $Z_k(u_k)$, and consequently can lead to different information gain.

Kullback-Leibler divergence

Kullback-Leibler divergence or cross-entropy is perhaps the well-known quantity in information theory and its applications. The Kullback-Leibler divergence of $p_0(x)$ and $p_1(x)$ is defined as [MD95; Kas97; Mah03b; vH14; Mah14]:

$$D_{KL}(p_0, p_1) = \int p_0(x) \log \frac{p_0(x)}{p_1(x)} \mathrm{d}x \tag{4.7}$$

In multi-object sensor management problem, the purpose of an optimal sensor control u_k at time k is to maximize the amount of information known about the target state x at time $k+1$; therefore, the Kullback-Leibler divergence of the prior density $\pi_{k|k}(x_k|Z_{1:k-1}, u_{0:k-1}, Z_k(u_k))$ from the predicted density $\pi_{k|k-1}(x_k|Z_{1:k-1}, u_{0:k-1})$ is defined as [Mah14]:

$$D_{KL}(\pi_{k|k}, \pi_{k|k-1}) = \int \pi_{k|k}(\cdot) \log \frac{\pi_{k|k}(\cdot)}{\pi_{k|k-1}(\cdot)} \mathrm{d}x \tag{4.8}$$

where:

$$\begin{aligned} \pi_{k|k}(\cdot) &= \pi_{k|k}(x_k|Z_{1:k-1}, u_{0:k-1}, Z_k(u_k)) \\ \pi_{k|k-1}(\cdot) &= \pi_{k|k-1}(x_k|Z_{1:k-1}, u_{0:k-1}) \end{aligned}$$

The predicted multi-object prior density $\pi_{k|k-1}(\cdot)$ represents the target state before the control action u_k, while the updated multi-object prior $\pi_{k|k}(\cdot)$ represents the target state following the control action u_k. The objective is to maximize the information gain by computing the Kullback-Leibler divergence between these two densities choosing the sensor control u_k that can lead to maximum information gain.

Cauchy-Schwarz divergence

The Kullback-Leibler divergence, and more generally the Rényi divergence, are the most used measures of information gain. However, their form make difficult analytic solutions for the probability density functions, even the most elementary ones, and in the case of more complicated pdfs, approximate numerical methods must be used. To address this problem, Hoang, Ba-Ngu Vo, Ba-Tuong, Mahler and Beard proposed a new intuitive and tractable objective function based on the Cauchy-Schwarz information functional [Mah14; Hoa+15; Bea+15].

The Cauchy-Schwarz divergence between the probability densities $p_0(x)$ and $p_1(x)$ of two point processes with respect to the reference measure z is defined by:

$$D_{CS}(p_0(x), p_1(x)) = -\log \frac{\langle p_0(x), p_1(x) \rangle_z}{||p_0(x)||_z \; ||p_1(x)||_z} \tag{4.9}$$

where $||f|| \overset{\text{def}}{=} \sqrt{\langle f, f \rangle}$ is the L^2-norm, and $\langle f, g \rangle \overset{\text{def}}{=} \int f(x)g(x)\mathrm{d}x$ is the inner product.

As shown by Hoang, the Cauchy-Schwarz divergence of two Poisson Random Finite Set (RFS) is half the squared distance between their densities. In addition, an analytical expression for the Cauchy-Schwarz divergence between two Poisson RFS was derived by Hoang and can be written as follows [Mah14; Hoa+15; Bea+15]:

$$D_{CS}(\pi_{k|k}, \pi_{k|k-1}) = -\log \frac{\int \pi_{k|k}(\cdot)\, \pi_{k|k-1}(\cdot)\, \mathrm{d}x}{\sqrt{\int \pi_{k|k}(\cdot)^2\, \pi_{k|k-1}(\cdot)^2\, \mathrm{d}x}} \tag{4.10}$$

where:

$$\pi_{k|k}(\cdot) = \pi_{k|k}(x_k | Z_{1:k-1}, u_{0:k-1}, Z_k(u_k))$$
$$\pi_{k|k-1}(\cdot) = \pi_{k|k-1}(x_k | Z_{1:k-1}, u_{0:k-1})$$

4.2 Risk-based Sensor Management

Risk-based methods aim to explicitly take into account the operational goals of a radar system by allocating the radar resources according to the risk posed to mission success. In this approach, the notion of operational risk is used for performing sensor management; hence, the name Risk-based sensor management. The Risk-based approach to sensor management appeared as a result of attempts to a) consider quantities that are directly of interest to the operational goal of the system, and b) obtain better situational awareness within a given operational context.

Risk-based sensor management can also be categorized under the heuristic or rules-based approaches. Following the rules-based approache, more sensor resources can be allocated to targets that are considered to be more threatening to mission while risk-based sensor management is considered a separate class due to the novelty of considering higher-level quantities that are defined according to the operational context.

The main advantage of Risk-based sensor management is that it explicitly takes into account the operational goal of a system. The disadvantage is that it does not focus on managing the uncertainty in the quantities of interest such that an operator can make decisions under lower uncertainty. To overcome this, priorities can be assigned to tasks or the reduction of operational risk can be considered [PR07]. Risk cannot be reduced by the sensor itself but rather with the use of an actuator.

To minimize the expected risk reduction, a decision is made on target classification. If it is incorrect, it results in a cost, i.e., a loss of a target of interest or a loss of sensor resources.

In this thesis, an incorrect decision is tagged as a type 1 error during statistical hypothesis testing. A type 1 error corresponds to an incorrect rejection of a true null hypothesis H_0, and occurs when it is true yet rejected.

The matrix CM_1, as defined by Equation 4.11, contains the cost of committing a type 1 error. Each column represents the true classification, and each row stands for the decision on a classification. The diagonal is zero since there is no cost when the correct decision is made.

$$CM_1 = \begin{array}{c} \\ 1 \\ 2 \\ \vdots \\ n \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & \dots & n \end{array} \\ \begin{pmatrix} 0 & c_{1_{12}} & \dots & c_{1_{1n}} \\ c_{1_{21}} & 0 & \dots & c_{1_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ c_{1_{n1}} & c_{1_{n2}} & \dots & 0 \end{pmatrix} \end{array} \tag{4.11}$$

where each $c_{1_{ij}}$ entry occurs when a decision falsely rejects H_0, resulting in a type 1 error.

4.2.1 Expected cost of committing a type 1 error

The expected cost of making a type 1 error when deciding on the classification of a target track is influenced by many factors, including the current classification, the probability of the actual target being lost (or not), and the decision about this classification. The above-mentioned factors are modeled by random variables. The expected cost can be obtained by applying the law of total expectation, as described in [Mar15] and detailed below.

Let C_1 be a discrete random variable representing the cost of the type 1 error. The cost matrix CM_1 contains entries $\{c_{1_{ij}}\}$ where each $c_{1_{ij}}$ entry occurs when a decision falsely rejects H_0, resulting in a type 1 error. J corresponds to a categorical random variable representing the current classification, $\{j|j \in [1,n]\}$. I is a categorical random variable denoting the decision on a classification, $\{i|i \in [1,n]\}$. $\hat{I}$ is a discrete, uniformly distributed, random variable denoting the classification decision on a reacquired target after it has been lost, $\{i|i \in [1,n]\}$. L is a Bernoulli random variable representing whether or not the actual target is lost, where the event space is $\{0,1\}$. We use the law of the iterated expectation for each random variable that determines the expected cost described as:

$$\begin{aligned} \mathbb{E}_{c_1}(C_1|I=i) &= \mathbb{E}_{c_1}(C_1|I=i,L=1,J=i)P(L=1)P(J=i) \\ &+ \mathbb{E}_{c_1}(C_1|I=i,L=1,J\neq i)P(L=1)P(J\neq i) \\ &+ \mathbb{E}_{c_1}(C_1|I=i,L=0,J=i)P(L=0)P(J=i) \\ &+ \mathbb{E}_{c_1}(C_1|I=i,L=0,J\neq i)P(L=0)P(J\neq i) \\ \\ &= \mathbb{E}_{c_1}(C_1|I=i,L=1,J=i)P(L=1)P(J=i) \\ &+ \mathbb{E}_{c_1}(C_1|I=i,L=1,J\neq i)P(L=1)P(J\neq i) \\ &+ 0 \\ &+ \mathbb{E}_{c_1}(C_1|I=i,L=0,J\neq i)P(L=0)P(J\neq i) \end{aligned} \tag{4.12}$$

In the above summation, the first term is the expected cost resulting from a situation in which a correct decision is made but the target is lost. The second term represents a wrong

decision and a loss of a target. The third term becomes zero since the decision is correct and the target is not lost. As for the fourth term, one can observe that a target is not lost in spite of a wrong decision being made. At this point, it would be necessary to consider the case in which the target would be reacquired. Note that regardless of the classifier's accuracy, it is possible that the acquired target is not the original one. Thus, (4.12) would be as follows:

$$\begin{aligned}
\mathbb{E}_{c_1}(C_1|I=i) &= \mathbb{E}_{c_1}(C_1|I=i,L=1,J=i,\hat{I}=i)P(L=1)P(J=i)P(\hat{I}=i)\\
&+\mathbb{E}_{c_1}(C_1|I=i,L=1,J=i,\hat{I}\neq i)P(L=1)P(J=i)P(\hat{I}\neq i)\\
&+\mathbb{E}_{c_1}(C_1|I=i,L=1,J\neq i,\hat{I}=i)P(L=1)P(J\neq i)P(\hat{I}=i)\\
&+\mathbb{E}_{c_1}(C_1|I=i,L=1,J\neq i,\hat{I}\neq i)P(L=1)P(J\neq i)P(\hat{I}\neq i)\\
&+\mathbb{E}_{c_1}(C_1|I=i,L=0,J\neq i)P(L=0)P(J\neq i)\\
&\\
&=0\\
&+\mathbb{E}_{c_1}(C_1|I=i,L=1,J=i,\hat{I}\neq i)P(L=1)P(J=i)P(\hat{I}\neq i)\\
&+\mathbb{E}_{c_1}(C_1|I=i,L=1,J\neq i,\hat{I}=i)P(L=1)P(J\neq i)P(\hat{I}=i)\\
&+0\\
&+\mathbb{E}_{c_1}(C_1|I=i,L=0,J\neq i)P(L=0)P(J\neq i) \qquad (4.13)
\end{aligned}$$

In the above summation, the first term becomes zero once the correct decision has been made, even though the track has been lost and later reacquired. Hence, there is no cost. In the second term, the cost is observed when the target is lost and reacquired and a wrong decision about its classification is made. In the third term, the cost is present and the target classification decision is never correct, even after the target has been lost and reacquired. The fourth term of summation represents the case in which the classification decision is incorrect (i.e. $I=i$ and $J\neq i$), and the reacquired target is characterized by a different classification (i.e., $\hat{I}\neq i$ and $J\neq i$). Consequently, the cost is zero because it does not exist with regard to the initial decision of $I=i$ before the track was lost. The last term illustrates the cost stemming from a wrong target classification since the target is never lost. These terms are related to specific rows and columns of the cost matrix CM_1 as shown below:

$$\begin{aligned}
\mathbb{E}_{c_1}(C_1|I=i) &= \mathbb{E}_{c_1}(C_1|I=i,L=1,J=i,\hat{I}\neq i)P(L=1)P(J=i)P(\hat{I}\neq i)\\
&+\mathbb{E}_{c_1}(C_1|I=i,L=1,J\neq i,\hat{I}=i)P(L=1)P(J\neq i)P(\hat{I}=i)\\
&+\mathbb{E}_{c_1}(C_1|I=i,L=0,J\neq i)P(L=0)P(J\neq i)\\
&\\
&=\sum_{c_{1_{ij}}} c_{1_{ij}}\Big(P(C_1=c_{1_{ij}}|I=i,L=1,J=i,\hat{I}\neq i)P(L=1)P(J=i)P(\hat{I}\neq i)\Big)\\
&+\sum_{c_{1_{ij}}} c_{1_{ij}}\Big(P(C_1=c_{1_{ij}}|I=i,L=1,J\neq i,\hat{I}=i)P(L=1)P(J\neq i)P(\hat{I}=i)\Big)\\
&+\sum_{c_{1_{ij}}} c_{1_{ij}}P(C_1=c_{1_{ij}}|I=i,L=0,J\neq i)P(L=0)P(J\neq i) \qquad (4.14)
\end{aligned}$$

$$\begin{aligned}\mathbb{E}_{c_1}(C_1|I=i) = &\sum_{r\in I} c_{1_{ri}} P(L=1)P(J=i)P(\hat{I}\neq i)\\ &+\sum_{r\in J} c_{1_{ir}} P(L=1)P(J=r)P(\hat{I}=i)\\ &+\sum_{r\in J} c_{1_{ir}} P(L=0)P(J=r) \qquad \forall r\neq i\end{aligned} \tag{4.15}$$

In Equation 4.15, the cost of the type 1 error C_1 for all $r = i$ is zero since there is no cost when the correct decision is made. Note that the first term in (4.15) is a function of the rows of the cost matrix over column $J = i$. This implies an incorrect decision was made after the target was reacquired. Finally, assuming $\hat{I}$ is uniformly distributed, and P_{lost} is the probability of the actual target being lost, (4.15) can be rewritten as below:

$$\begin{aligned}\mathbb{E}_{c_1}(C_1|I=i) = &\sum_{r\in I} c_{ri} P(J=i) P_{lost}\frac{n-1}{n}\\ &+\sum_{r\in J} c_{ir} P(J=r) P_{lost}\frac{1}{n}\\ &+\sum_{r\in J} c_{ir} P(J=r)(1-P_{lost}) \quad \forall r\neq i\end{aligned} \tag{4.16}$$

The probability P_{lost} is assumed to be the portion of a multivariate normal distribution $\mathcal{N}(\hat{x}_k, \hat{P}_k)$ not contained in the sensor's FOV when the sensor's aim-point is centered on the kinematic state of the target ($\hat{x}$ is the mean state estimate and $\hat{P}$ is the state estimate covariance). The expected cost obtained in (4.16) was introduced by Martin [Mar15] based on works of Papageorgiou and Wang [PR07; WHE11].

4.2.2 The Expected Risk Reduction (ERR) metric

When a decision on a target classification is made, the goal is to minimize the risk, so the minimum expected cost is chosen among all possible decisions for each track classification. The risk always decreases with new measurements, reducing the probability of the target being misclassified or lost [PR07; WHE11; Mar15; Gom+16].

The Expected Risk Reduction (ERR) is achieved using the minimum expected cost presented in (4.16). Note that the probabilities in this equation change as measurements are accumulated by a sensor. It is assumed that these probabilities change as a Bayesian update. Denoting R as the minimum cost before a measurement update, we can calculate the Expected Risk Reduction (ERR) as:

$$R_i \triangleq \mathbb{E}_{c_1}(C_1|I=i) \tag{4.17}$$

$$R = \min_i\{R_i\} \tag{4.18}$$

Assuming that the posterior probabilities are denoted by P'_{lost} and $P'(J=i)$, the risk using these updated probabilities is:

$$R' = \min_i\{R'_i\} = \min_i \left\{ \begin{array}{ll} & \sum\limits_{r\in I} c_{ri} P'(J=i) P'_{lost}\frac{n-1}{n} \\ + & \sum\limits_{r\in J} c_{ir} P'(J=r) P'_{lost}\frac{1}{n} \\ + & \sum\limits_{r\in J} c_{ir} P'(J=r)(1-P'_{lost}) \end{array} \right\} \tag{4.19}$$

When the classification probability is updated through the direct application of Bayes' theorem, then (4.19) can be rewritten as follows:

$$R' = \min_i\{R'_i\} = \min_i \left\{ \begin{array}{ll} & \sum\limits_{r\in I} c_{ri} \frac{P(M=m|J=i)P(J=i)}{P(M=m)} P'_{lost}\frac{n-1}{n} \\ + & \sum\limits_{r\in J} c_{ir} \frac{P(M=m|J=r)P(J=r)}{P(M=m)} P'_{lost}\frac{1}{n} \\ + & \sum\limits_{r\in J} c_{ir} \frac{P(M=m|J=r)P(J=r)}{P(M=m)} (1-P'_{lost}) \end{array} \right\} \tag{4.20}$$

Since any classification measurement M is possible, it is necessary to calculate an additional expectation of all possible measurements $\langle R' \rangle$ that can be illustrated as:

$$\begin{aligned} \langle R' \rangle &= \sum_{m\in M} R' P(M=m) \\ &= \sum_{m\in M} \min_i \left\{ \begin{array}{ll} & \sum\limits_{r\in I} c_{ri} \left(P(M=m|J=i)P(J=i)P'_{lost}\frac{n-1}{n} \right) \\ + & \sum\limits_{r\in J} c_{ir} \left(P(M=m|J=r)P(J=r)P'_{lost}\frac{1}{n} \right) \\ + & \sum\limits_{r\in J} c_{ir} \left(P(M=m|J=r)P(J=r)(1-P'_{lost}) \right) \end{array} \right\} \end{aligned} \tag{4.21}$$

Taking into consideration that the expected cost decreases in value with new measures [PR07; Mar15; Gom+16], the radar-beam direction based on ERR is given as follows:

$$u_k = \arg\max_{u\in\mathbb{U}_k} \left[ERR(u) \right] \tag{4.22}$$

where $u = \arctan(x^{(j)}/y^{(j)})$, $\forall j \in [1, \cdots, N_{targets}]$ is the sensor control, and $ERR(u) = R^{(j)} - \langle R'^{(j)} \rangle$. The sensor control that provides the greatest reduction in this ERR value is chosen to take the actual measurement.

4.3 ERR numerical example using the EKF filter

This example introduces the Expected Risk Reduction (ERR) approach to sensor management and multi-target tracking in a surveillance context that implements an IR-Radar sensor system. Due to operational restrictions (for instance, electromagnetic emission constraints), it is assumed that there are more targets than given sensors are capable of tracking simultaneously when a radar emission control is applied. It is also presumed that an incorrect target classification entails a cost that is different for each target class. The ERR is then applied to a simulated IR-Radar sensor management to preserve an acceptable level of kinematic accuracy on targets of high cost. The task of a sensor manager is to decide which targets the sensor should focus on in order to reduce the expected cost of an incorrect classification decision. Finally, empirical statistical tests show that a track on high priority targets is maintained better when the aforementioned approach is introduced than in the case of other conventional methods, such as the information-theoretic sensor management or the round robin assignment.

In order to evaluate the performance of the sensor management processing the ERR metric, we run a set of two scenarios. Each concerns a 300-second duration. The first scenario involves 10 maneuvering targets, 4 of which are targets of interest (targets 2, 3, 8, and 10) and consequently should be tracked. The second scenario embraces 15 maneuvering targets, 4 of which are targets of interest (targets 3, 5, 12, and 15). The ground truth over 300 seconds for scenario 1 and 2 is shown in Figure 4.2 and Figure 4.3.

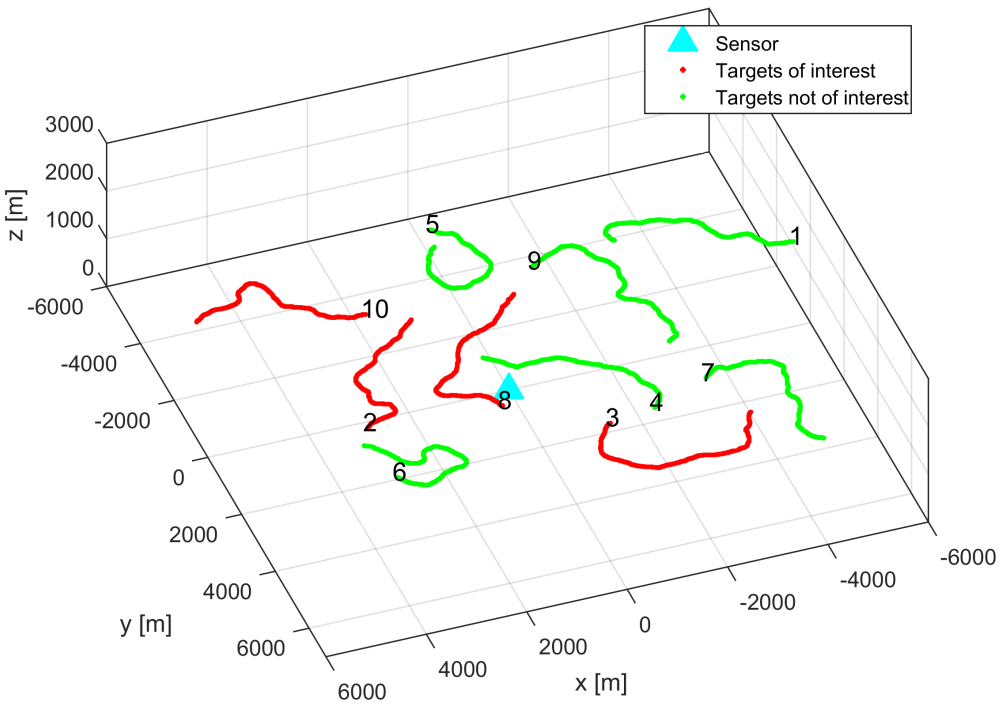

Figure 4.2: Scenario 1: targets of interest (in red), targets not of interest (in green), and the sensor location (cyan triangle).

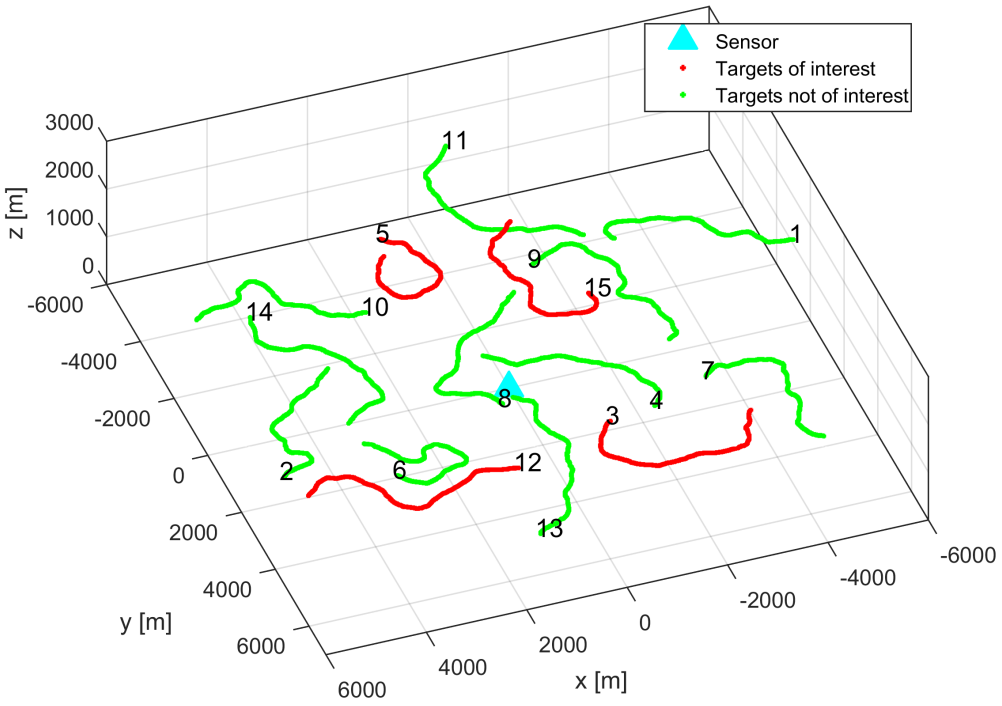

Figure 4.3: Scenario 2: targets of interest (in red), targets not of interest (in green), and the sensor location (cyan triangle).

The sensor system combines an Infrared Search and Track (IRST) and a Radar. Both are located at the reference position ($x = y = z = 0$). The IR sensor obtains measurements of all targets every second, while the radar sensor takes a measurement of only one target every 2 seconds (using electromagnetics emission constraints). Thus, every 2 seconds a sensor management algorithm based on the risk metrics decides which target track to estimate using the radar sensor report. The radar sensor field of view (FOV) is a 500 m^2 region centered on the estimated track position. The target is considered lost if the ground truth position is outside of this FOV. In such a case, a state estimate is very poor and no further measurements are made on targets.

For each target, there is information about the kinematic true state and its classification, and both are represented by X as follows:

$$X = [X_{kinematic} \quad X_{classification}]$$

The varying turn rate of a nearly constantly turning kinematic model (see Appendix D) is considered [BLK01; LJ03]. Thus, the true kinematic state consists of a three dimensional position, velocity and turn rate. An Extended Kalman filter (EKF), as introduced in Chapter 1.1.3, is employed to estimate the position, velocity, and acceleration of a target. Both targets and sensors are defined in a three dimensional space. Consequently, the true kinematic state consists of a three dimensional position and velocity:

$$X_{kinematic} = \begin{bmatrix} x \\ \dot{x} \\ y \\ \dot{y} \\ z \\ \dot{z} \end{bmatrix}$$

IR measurements basically provide angular information, an azimuth angle θ and an elevation angle ϕ, whereas radar measurements consist of an azimuth angle θ, an elevation angle ϕ, a range r, and range rate $\dot{r}$ as shown hereafter:

$$Z_{IR} = \begin{bmatrix} \theta \\ \phi \end{bmatrix} \qquad Z_{Radar} = \begin{bmatrix} \theta \\ \phi \\ r \\ \dot{r} \end{bmatrix}$$

where:

$$\theta = \arctan\left(\frac{y}{x}\right) \qquad r = \sqrt{x^2 + y^2 + z^2}$$
$$\phi = \arctan\left(\frac{z}{\sqrt{x^2 + y^2}}\right) \qquad \dot{r} = \frac{(x\dot{x} + y\dot{y} + z\dot{z})}{r}$$

To avoid singularities in the linearization process, measurements are converted to Cartesian coordinates. The linearization on the noise is achieved by following the description presented by Bar-Shalom [BWT11], and measurements are represented as:

$$Z_{IR}^{Cartesian} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \qquad Z_{Radar}^{Cartesian} = \begin{bmatrix} x \\ y \\ z \\ \dot{r} \end{bmatrix}$$

where,

$$x = r\cos(\phi)\cos(\theta), \qquad y = r\cos(\phi)\sin(\theta), \qquad z = r\sin(\phi)$$

It is assumed that the initial position of targets is known. The matrix H is given as shown below. Note that it corresponds here to a linear filter.

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

The measurement error covariance matrix R is defined as follows:

$$R = \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} & \Sigma_{xz} \\ \Sigma_{yx} & \Sigma_{yy} & \Sigma_{yz} \\ \Sigma_{zx} & \Sigma_{zy} & \Sigma_{zz} \end{bmatrix}$$

The subsequent paragraph demonstrates how to calculate the matrix R using linearization. Measures θ_m, ϕ_m, and r_m are defined with respect to true θ, ϕ, and r data as follows:

$$\begin{aligned} \theta_m &= \theta + \tilde{\theta} \\ \phi_m &= \phi + \tilde{\phi} \\ r_m &= r + \tilde{r} \end{aligned}$$

where errors $\tilde{\theta}, \tilde{\phi}$, and $\tilde{r}$ are assumed to be independent with zero mean and standard deviations $\sigma_\theta, \sigma_\phi, \sigma_r$ respectively. Denoting (x, y, z) the true Cartesian position and using first order terms of the Taylor series expansion of the Cartesian measurements at (θ_m, ϕ_m, r_m), i.e., applying linearization, yields the Cartesian coordinate errors as follows:

$$\begin{aligned} x_m - x &\approx \tilde{r}\cos(\phi)\cos(\theta) - \tilde{\phi}\, r_m \sin(\phi)\cos(\theta) - \tilde{\theta} r_m \cos(\phi)\sin(\theta) \\ y_m - y &\approx \tilde{r}\cos(\phi)\sin(\theta) - \tilde{\phi}\, r_m \sin(\phi)\sin(\theta) + \tilde{\theta} r_m \cos(\phi)\cos(\theta) \\ z_m - z &\approx \tilde{r}\sin(\phi) + \tilde{\phi}\, r_m \cos(\phi) \end{aligned}$$

The mean of the errors, as given by the above equations, is zero. Consequently, the elements of the corresponding covariance matrix R are:

$$\begin{aligned} \Sigma_{xx} &= \sigma_r^2 \cos^2(\phi)\cos^2(\theta) + \sigma_\phi^2 r^2 \sin^2(\phi)\cos^2(\theta) + \sigma_\theta^2 r^2 \cos^2(\phi)\sin^2(\theta) \\ \Sigma_{yy} &= \sigma_r^2 \cos^2(\phi)\sin^2(\theta) + \sigma_\phi^2 r^2 \sin^2(\phi)\sin^2(\theta) + \sigma_\theta^2 r^2 \cos^2(\phi)\cos^2(\theta) \\ \Sigma_{zz} &= \sigma_r^2 \sin^2(\phi) + \sigma_\phi^2 r^2 \cos^2(\phi) \\ \Sigma_{xy} &= \sigma_r^2 \cos^2(\phi)\cos(\theta)\sin(\theta) + \sigma_\phi^2 r^2 \sin^2(\phi)\cos(\theta)\sin(\theta) - \sigma_\theta^2 r^2 \cos^2(\phi)\cos(\theta)\sin(\theta) \\ \Sigma_{xz} &= \sigma_r^2 \cos(\phi)\sin(\phi)\cos(\theta) - \sigma_\phi^2 r^2 \cos(\phi)\sin(\phi)\cos(\theta) \\ \Sigma_{yz} &= \sigma_r^2 \cos(\phi)\sin(\phi)\sin(\theta) - \sigma_\phi^2 r^2 \cos(\phi)\sin(\phi)\sin(\theta) \end{aligned}$$

we set $\sigma_r^2 = 1$ m^2 and $\sigma_\theta^2 = \sigma_\phi^2 = 3.0462 \times 10^{-4}$ rad^2 (i.e. the standard deviation of 1 degree). The process noise covariance matrix Q is given by:

$$Q = \Phi_s \begin{bmatrix} \frac{T^4}{4} & \frac{T^3}{2} & 0 & 0 & 0 & 0 \\ \frac{T^3}{2} & T^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{T^4}{4} & \frac{T^3}{2} & 0 & 0 \\ 0 & 0 & \frac{T^3}{2} & T^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{T^4}{4} & \frac{T^3}{2} \\ 0 & 0 & 0 & 0 & \frac{T^3}{2} & T^2 \end{bmatrix}$$

The scan rate of the IR sensor is $T_{IR} = 1$ second, while for the Radar sensor $T_{Radar} = 2$ seconds, and $\Phi_s = 5$ m/s^2. Finally, the state transition model F is a basic constant velocity model:

$$F = \begin{bmatrix} 1 & T & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & T & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The classification state estimate is formulated below. Assuming that there are n possible classification states for each target, J is a random variable that stands for the true classification with support $\{j|j \in [1, n]\}$.

$$X_{classification} = \begin{bmatrix} P(J = 1) \\ \vdots \\ P(J = n) \end{bmatrix} \tag{4.23}$$

The classification probability is updated by applying the Bayes' theorem as shown below. The classification measurement is represented by a discrete random variable M with support $\{m|m \in [1, n]\}$.

$$\begin{aligned} P'(J = i) &\stackrel{\text{def}}{=} P(J = i|M = m) \\ &= \frac{P(M = m|J = i)P(J = i)}{P(M = m)} \\ &= \frac{P(M = m|J = i)P(J = i)}{\sum_{r=1}^{n} P(M = m|J = r)P(J = r)} \end{aligned} \tag{4.24}$$

where P' indicates the posterior probability. To simplify classification notations in this chapter, measurement likelihoods $P(M = m|J = i)$ are represented by a normalized confusion matrix CC.

$$CC = \begin{array}{c} \\ 1 \\ \vdots \\ n \end{array} \begin{array}{c} \begin{array}{ccc} 1 & \dots & n \end{array} \\ \begin{pmatrix} P(M = 1|J = 1) & \dots & P(M = 1|J = n) \\ \vdots & \ddots & \vdots \\ P(M = n|J = 1) & \dots & P(M = n|J = n) \end{pmatrix} \end{array}$$

For comparison purposes, the ERR approach was contrasted with three different sensor management methods including the Kullback-Leibler divergence (KL) divergence, the random assignment, and the round robin assignment. In the latter, the targets are repeatedly selected in a specific order. 1000 Monte Carlo runs were conducted using each method. The track error was calculated between the ground truth position and the estimate for all tracks. For each analysis below, the 5% of the highest and the lowest error measurements were discarded to remove outliers.

To calculate the KL divergence we used two multivariate normal distributions, with means μ_1, μ_2 and covariance matrices Σ_1, Σ_2. The two distributions have the same dimension, k. In this case, the KL divergence corresponds to:

$$D_{KL}(\mathcal{N}_1||\mathcal{N}_2) = \frac{1}{2}\left((\mu_2-\mu_1)^T\Sigma_2^{-1}(\mu_2-\mu_1) - k + trace(\Sigma_2^{-1}\Sigma_1) + \log\left(\frac{\det\Sigma_2}{\det\Sigma_1}\right)\right)$$

For evaluating ERR metric, a binary classification state is considered where the target to be tracked is either a target of interest ($J=1$) or a target not of interest ($J=2$). The binary classification measurement M has support $\{m|m\in[1,2]\}$. Cost matrix CM_1 and confusion matrix CC are:

$$CM_1 = \begin{array}{c} \\ 1 \\ 2 \end{array}\begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \begin{pmatrix} 0 & 1 \\ 30 & 0 \end{pmatrix} \end{array} \qquad CC = \begin{array}{c} \\ 1 \\ 2 \end{array}\begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \begin{pmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{pmatrix} \end{array}$$

4.3.1 Binary classification - known initial classification

In this case, all target classifications are initially known. Table 4.1 presents the resulting median error on each target for each sensor manager method in scenario 1.

Table 4.1: The median position error in meters for scenario 1. Targets of interest are 2, 3, 8, and 10.

Class	Target	ERR	KLdiv	Random	Round Robin
2	1	8949	8949	7943	107
1	2	27	122	1976	112
1	3	40	7191	5875	6215
2	4	4900	29	2365	46
2	5	7777	36	4605	62
2	6	6053	6053	3500	85
2	7	5778	5778	25001	88
1	8	31	40	3765	91
2	9	7848	7848	7600	120
1	10	41	6867	4218	105

For all targets of interest, the ERR approach maintains the value of an acceptable error that is lower when compared to others' methods. The sensor manager using ERR maintains a track on targets 2, 3, 8, and 10 with the error's value lower than the sensor's FOV radius (250m). The KL divergence has poor performance for target 3 and 10. The random method performs very poorly and the round robin method has poor performance for target 3.

When using ERR, more measurements can be assigned to targets 2, 3, 8, and 10, since they are targets of interest. Thus, tracking performance is improved yet the track quality for targets of no interest is diminished.

To examine the results presented by metrics more accurately, it is necessary to notice that the average track error is less than 250 meters, and therefore still within the sensor's FOV.

If the average error on targets 2, 3, 8, and 10 is greater than 250 meters, it means that the sensor management method was ineffective on average.

Table 4.2 focuses on p-values for a Student t-test (H_0: true mean error is $\leq$ 250 meters). We can observe that the ERR approach is the only method that effectively maintains track on all targets of interest.

Table 4.2: p-values for H_0: residual error $\leq$ 250 meters

Target	ERR	KLdiv	Random	Round Robin
2	1	1	0	1
3	1	0	0	0
8	1	1	0	1
10	1	0	0	1

Table 4.3 represents the resulting median error on each target for each sensor manager method in scenario 2. For all targets of interest, the ERR approach maintains track on targets 3, 5, 12, and 15. The error does not exceed sensor's FOV radius (250m). The KL divergence performs well only for target 15. The random method performs very poorly, and the round robin method performs well only for target 5.

Table 4.3: The median position error in meters for scenario 2. Targets of interest are 3, 5, 12, and 15.

Class	Target	ERR	KLdiv	Random	Round Robin
2	1	8949	8949	7880	8949
2	2	4965	4965	2826	132
1	3	226	7191	6336	6220
2	4	4900	44	3165	56
1	5	33	7777	5571	72
2	6	6053	6053	4274	104
2	7	5778	5778	6617	101
2	8	5497	68	4361	102
2	9	7848	7848	7416	115
2	10	6867	6867	5154	114
2	11	4841	4841	3539	105
1	12	39	7102	5280	7102
2	13	7657	32	5694	7657
2	14	6913	6913	4733	124
1	15	34	35	4992	6690

Table 4.4 provides p-values for a Student t-test (H_0: true mean error is $\leq$ 250 meters).

Table 4.4: p-values for H_0: residual error $\leq$ 250 meters

Target	ERR	KLdiv	Random	Round Robin
3	1	0	0	0
5	1	0	0	1
12	1	0	0	0
15	1	1	0	0

4.3.2 Binary classification - unknown initial classification

In this case, all target classifications are initially unknown. Table 4.5 provides the resulting median error on each target for each sensor manager method in scenario 1. For all targets of interest, the Expected Risk Reduction (ERR) approach and the Kullback-Leibler divergence (KL) divergence perform well for targets 2 and 8. The random method performs very poorly, and the round robin method only fails to track target 3.

Table 4.5: The median position error in meters for scenario 1. Targets of interest: 2, 3, 8, and 10. All target classifications are initially unknown.

Class	Target	ERR	KLdiv	Random	Round Robin
2	1	8487	8949	8029	107
1	2	30	122	1980	112
1	3	7191	7191	6082	6216
2	4	1578	29	2440	46
2	5	1748	36	4369	62
2	6	1405	6053	3382	85
2	7	2539	5778	22672	88
1	8	34	40	3703	91
2	9	2749	7848	7981	120
1	10	3269	6867	4468	105

Table 4.6 shows the resulting median error on each target for each sensor manager method in scenario 2 where all target classifications are initially unknown. For all targets of interest, the ERR approach maintains tracks on targets 5 and 15 with the error not exceeding the sensor's FOV radius (250m). The KL divergence performs well only for target 15. The random method performs very poorly, and the round robin method performs well only for target 5.

Table 4.6: Median position error in meters for scenario 2. Targets of interest are 3, 5, 12, and 15. Initially, all target classifications are unknown.

Class	Target	ERR	KLdiv	Random	Round Robin
2	1	8597	8949	7880	8949
2	2	637	4965	2826	132
1	3	7191	7191	6336	6220
2	4	571	44	3165	56
1	5	54	7777	5571	72
2	6	5676	6053	4274	104
2	7	3825	5778	6617	101
2	8	1057	68	4361	102
2	9	7848	7848	7416	115
2	10	6867	6867	5154	114
2	11	1516	4841	3539	105
1	12	7102	7102	5280	7102
2	13	2539	32	5694	7657
2	14	3832	6913	4733	124
1	15	138	35	4992	6690

4.3.3 Ternary classification - unknown initial classification

In order to evaluate the ERR metric, we also consider a ternary classification state where the target being tracked is either a target of high interest ($J = 1$), of medium interest ($J = 2$), or of low interest ($J = 3$). The ternary classification measurement M has support $\{m|m \in [1, 2, 3]\}$. Cost matrix CM_1 and confusion matrix CC are:

$$CM_1 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{pmatrix} 0 & 20 & 1 \\ 30 & 0 & 1 \\ 30 & 20 & 0 \end{pmatrix} \end{array} \qquad CC = \begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{pmatrix} 0.8 & 0.1 & 0.1 \\ 0.1 & 0.8 & 0.1 \\ 0.1 & 0.1 & 0.8 \end{pmatrix} \end{array}$$

Initially, all target classifications are unknown. Table 4.7 shows the resulting median error on each target for each sensor manager method for scenario 1. For all targets of interest, the ERR approach performs well only for target 2. The KL divergence performs well for targets 2 and 8. The random method performs very poorly, and the round robin method only fails to track target 3. While comparing Table 4.5 and 4.7, we can observe that the ERR method improves results in the ternary classification on target 10 (hight interest) despite the opposite impact on targets 2 and 8 (medium interest).

Table 4.7: The median position error in meters for scenario 1. There are the following targets of interest: 2, 8, and 10. Initially, all target classifications are unknown.

Class	Target	ERR	KLdiv	Random	Round Robin
3	1	8704	8949	7909	107
2	2	47	122	2182	112
3	3	7191	7191	5543	6220
3	4	2112	29	2228	46
3	5	1994	36	3600	62
3	6	4193	6053	3346	85
3	7	5660	5778	10838	88
2	8	293	40	4290	91
3	9	2716	7848	7616	120
1	10	1825	6867	3587	105

Table 4.8 illustrates the resulting median error on each target for each sensor manager method in scenario 2. For all targets of interest, the ERR approach performs well only in the case of target 5, and shows that the value of the average error on targets 12 and 15 is lower than when other methods are employed, but higher than the sensor's FOV. The KL divergence performs well only for target 8. The random method performs very poorly, and the round robin method can only track target 5.

If we compare Table 4.6 and 4.8, we notice improved results for the ternary classification of tracks on target 12 when the ERR method is applied. As for tracks on targets 5 and 15 (medium interest), the outcomes are less satisfactory. However, the value of an average error on target 12 turns out to be higher than the sensor's FOV.

Table 4.8: The median position error in meters for scenario 2. There are the following targets of interest: 5, 12, and 15. Initially, all target classifications are unknown.

Class	Target	ERR	KLdiv	Random	Round Robin
3	1	8251	8949	7823	8949
3	2	2174	4965	2495	132
3	3	7191	7191	6205	6211
3	4	1890	44	2765	56
2	5	157	7777	5230	72
3	6	6053	6053	4324	104
3	7	5778	5778	22830	101
3	8	2438	67	4193	102
3	9	7848	7848	7209	115
3	10	6867	6867	5104	114
3	11	3727	4841	3656	105
1	12	2095	7102	5119	7102
3	13	7657	32	5950	7657
3	14	6913	6913	4822	124
2	15	308	35	5208	6690

4.4 ERR numerical example using the GM-PHD Tracker

In order to demonstrate the proposed approach we use a numerical example where a multi-function radar (MFR) is controlled to track an unknown number of targets. It is able to track targets in the sector defined by [0, 2000] meters in range and $[-\pi/2, \pi/2]$ rad in bearing using its "pencil" beam. The radar has a 4-degree beamwidth. The true trajectories are shown in Figure 4.4. The duration of the scenario is 300 seconds.

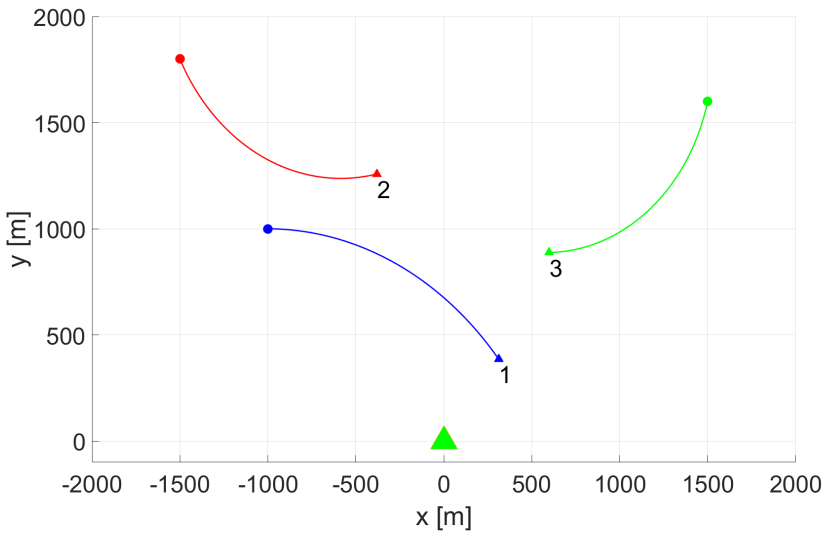

Figure 4.4: The trajectories of three observed targets. The radar is at the origin of the axes. Start and stop positions for each track are shown with ∘ and △.

We take into consideration a surveillance context with electromagnetic emission constraints as presented in [ML99]. It is assumed that there are too many maneuverable targets to be tracked by the radar system. Only a subset of all the targets need to be tracked and, initially, we have knowledge of their kinematic states, yet their classification states are un-

known. Finally, it is also presumed that there is a cost resulting from an incorrect decision of a target's true classification. Thus, the task of a sensor manager is to decide which targets the sensor should focus on in order to reduce the expected cost of an incorrect classification decision.

For each target, there is information about the kinematic true state and its classification. Both data are represented by X as follows:

$$X = [X_{kinematic} \quad X_{classification}]$$

The varying turn rate of a nearly constantly turning kinematic model (see Appendix D) is considered [LJ03]. Thus, the true kinematic state consists of a two dimensional position, velocity and turn rate:

$$X_{kinematic} = [x, \dot{x}, y, \dot{y}, \omega]^T$$

The transition model in target tracking can be formulated as follows:

$$\begin{aligned} x_k &= F(\omega_{k-1})x_{k-1} + Gw_{k-1} \\ \omega_k &= \omega_{k-1} + Tu_{k-1} \end{aligned}$$

where,

$$F(\omega) = \begin{bmatrix} 1 & \frac{\sin \omega T}{\omega} & 0 & -\frac{1-\cos \omega T}{\omega} \\ 0 & \cos \omega T & 0 & -\sin \omega T \\ 0 & \frac{1-\cos \omega T}{\omega} & 1 & \frac{\sin \omega T}{\omega} \\ 0 & \sin \omega T & 0 & \cos \omega T \end{bmatrix} \qquad G = \begin{bmatrix} \frac{T^2}{2} & 0 \\ T & 0 \\ 0 & \frac{T^2}{2} \\ 0 & T \end{bmatrix}$$

$T = 1s$ is the sampling period; $w_{k-1} \sim \mathcal{N}(\cdot; 0, \sigma_w^2 I)$ and $u_{k-1} \sim \mathcal{N}(\cdot; 0, \sigma_u^2 I)$ are the process noise with standard deviation $\sigma_w = 15m/s^2$ and $\sigma_u = (\pi/180)rad/s$. The targets are observed via a radar that provides range and bearing measurements. Each target is detected with probability $p_{D,k} = 0.98$, and the measurement uses the observation model given by:

$$z_k = \begin{bmatrix} \arctan(x/y) \\ \sqrt{x^2 + y^2} \end{bmatrix} + \varepsilon_k$$

where $\varepsilon_k \sim \mathcal{N}(\cdot; 0, R_k)$; $R_k = diag([\sigma_\theta^2, \sigma_r^2])$ is the measurement noise covariance matrix, with $\sigma_\theta = (0.5\pi/180)rad$, and $\sigma_r = 10m$. The detected measurements are immersed in clutter, which is typically modeled as a Poisson RFS with intensity function:

$$\kappa_k(z) = \lambda_c V u(z)$$

where $u(\cdot)$ represents the uniform density over the surveillance region, $V = 3.14 \times 10^5 (radm)$ is the area of the surveillance region, and $\lambda_c = 3.18 \times 10^{-5} (radm)^{-1}$ is the average clutter intensity. In that case, an average of 10 clutter points per scan is received with

$P_{FA} = 0.10$ at each time instance. The classification state estimate is formulated similarly to the previous example in Section 4.3, using the Extended Kalman filter (EKF). Assuming that there are n possible classification states for each target, J is a random variable that stands for the true classification with support $\{j|j \in [1, n]\}$.

$$X_{classification} = \begin{bmatrix} P(J=1) \\ \vdots \\ P(J=n) \end{bmatrix}$$

The classification probability is updated by applying the Bayes' theorem (4.24). The radar has a 4-degree beamwidth. If the target ground truth corresponding to the track position is outside this FOV (i.e. the state estimate is very poor), the track is considered lost. The probability P_{lost} is assumed to be the portion of a multivariate normal distribution $\mathcal{N}(\hat{x}_k, \hat{P}_k)$ not contained in the sensor's FOV when the sensor's aim-point is centered on a kinematic state of the target ($\hat{x}$ is the mean state estimate and $\hat{P}$ is the state estimate covariance).

In this chapter we use the Gaussian-Mixture PHD Tracker (GM-PHDT) to propagate a parametrized approximation of the multi-target posterior by applying the gating and pruning/merging procedures (see Chapter 1.3.1). Gating is performed at each time step using a 99% validation gate (the region centered on the predicted measurement with a 0.99 probability of containing a primary object generated measurement), as described in [BP99; BWT11]. Pruning and merging are performed at each time step using a weight threshold of $T_{th} = 10^{-5}$ and a merging threshold of $U = 4m$, see [PVC06].

For comparison purposes, the ERR approach was contrasted with three different sensor management methods involving the Posterior Expected Number of Targets (PENT), the Posterior Expected Number of Targets of Interest (PENTI), and the random assignment. 1000 Monte Carlo runs were conducted using each method. All target classifications are initially unknown and each target starts with a high accuracy kinematic track. Therefore, the main tasks of the radar are to correctly classify and track the targets, and allocate measurements to the target of interest. Tracking accuracy is measured using the Optimal SubPattern Assignment (OSPA). The OSPA, presented in Chapter 1.2.9, measures the error between the true and the estimated multi-target states, X_k and $\hat{X}_k$, respectively. Two simulation examples are used to test the proposed approach to sensor management. Additional examples can be found in[Gom+16; Gom+17].

4.4.1 Binary classification

In order to evaluate the Expected Risk Reduction (ERR) metric, a binary classification state is considered where the target to be tracked is either a target of interest ($J = 1$) or a target not of interest ($J = 2$). The binary classification measurement M has support $\{m|m \in [1, 2]\}$. Cost matrix CM_1 and confusion matrix CC are:

$$CM_1 = \begin{array}{c} \\ 1 \\ 2 \end{array} \begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \begin{pmatrix} 0 & 1 \\ 30 & 0 \end{pmatrix} \end{array} \qquad CC = \begin{array}{c} \\ 1 \\ 2 \end{array} \begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \begin{pmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{pmatrix} \end{array}$$

In this example, targets 2 and 3 are those of interest. Figure 4.5 shows a typical run of the proposed algorithm at the instant $k = 40$ seconds.

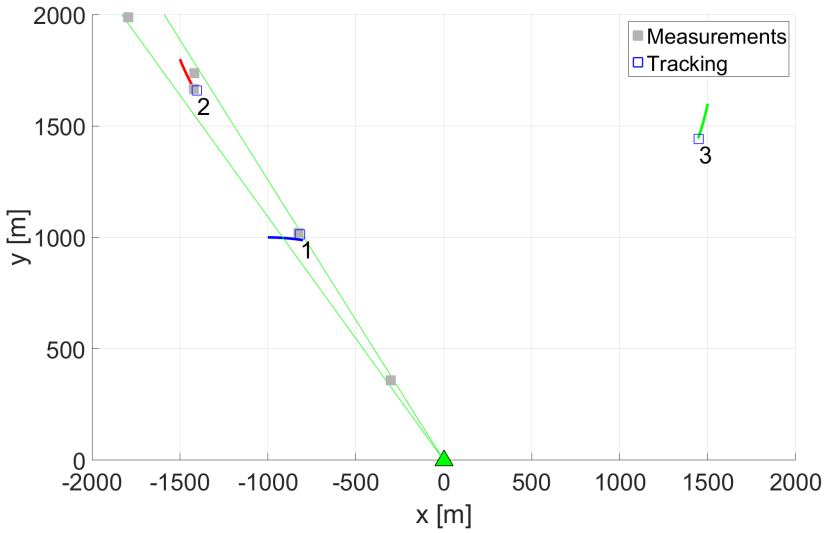

Figure 4.5: Typical run to the ERR approach at instant $k = 40$ seconds. The radar is at the origin of the axes. Target tracking and identification are provided using the GM-PHD tracker.

The ERR was compared with the PENTI and the random sensor management. On all three sensor management approaches, the OSPA metric, along with the cardinality estimation and localization error, is averaged over 1000 independent Monte Carlo runs. In Figure 4.6, the Monte Carlo average of the OSPA distance for $p = 1$ and $c = 100m$ is shown. It can be observed that the curves stabilize to an average error close to $60m$ per target for the PENTI and the random schemes, and close to $50m$ per target for the ERR approach.

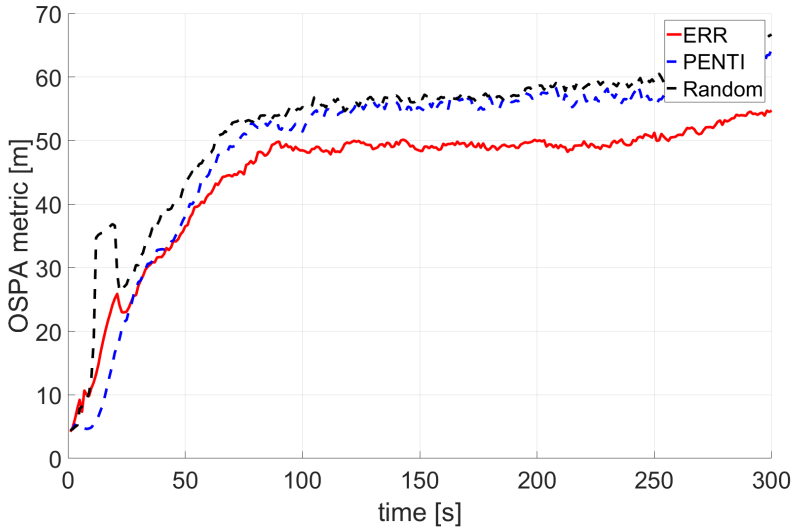

Figure 4.6: OSPA metric. Error performance of the three sensor management approaches with target prioritization, averaged over 1000 Monte Carlo runs.

Examining the cardinality estimation and the localization error given in Figures 4.7 and 4.8, it can be seen that in terms of localization error, the PENTI and ERR sensor management settle to an error consistent with the standard deviation of the measurement noise, while the random schedule achieves an unexpectedly lower localization error.

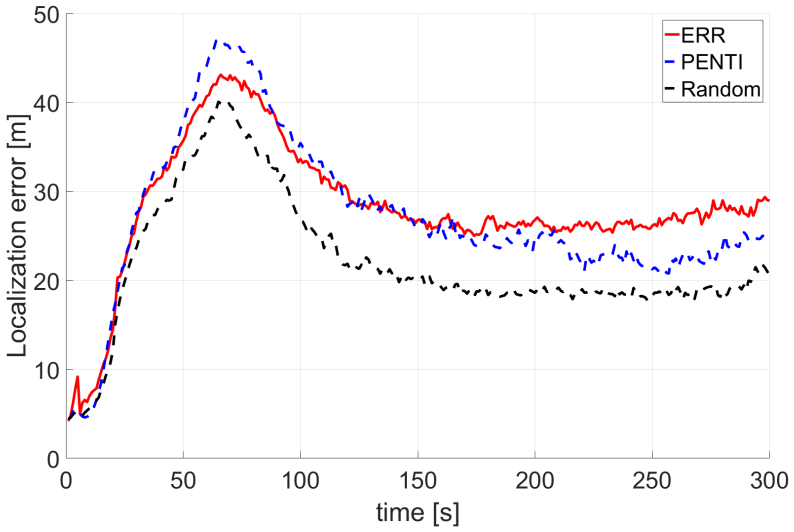

Figure 4.7: Localization error performance of the three sensor management approaches with target prioritization, averaged over 1000 Monte Carlo runs.

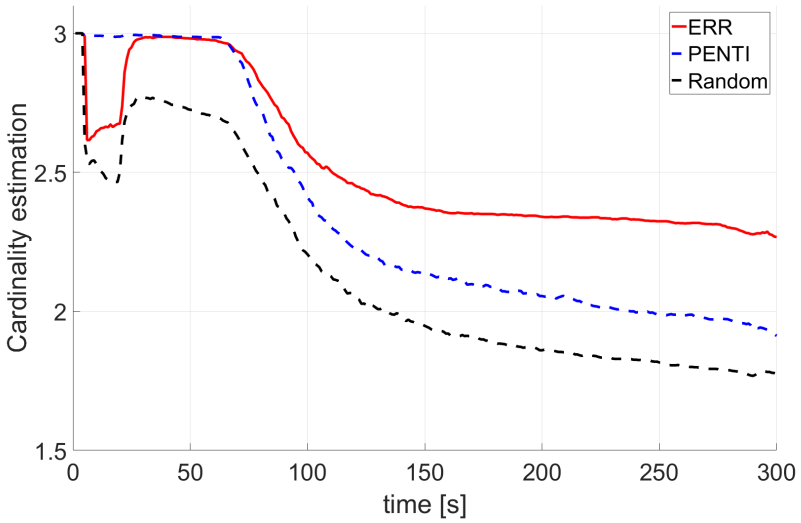

Figure 4.8: Estimated number of targets for three sensor management approaches with target prioritization, averaged over 1000 Monte Carlo runs.

Analyzing Figure 4.8, we can observe that, on average, the random method can keep track only of 2 targets, while the ERR method is capable of tracking about 2.5 targets. The ERR approach is penalized much more than the PENTI and the random scheme since it is more successful at keeping track of more targets at the expense of the localization estimation, which is less accurate. In fact, in terms of cardinality estimation, the ERR scheme is at a significant advantage in this scenario.

4.4.2 Mono classification

In order to evaluate the ERR metric without prioritization, a mono classification state is considered. In that case, each target belongs to precisely the same class ($J = 1$), and targets 1, 2, and 3 should be tracked with the same priority. In this example, the ERR is compared with the PENT and the random sensor management. The OSPA metric, as well as the cardinality estimation and localization error of all three sensor management approaches, are averaged over 1000 independent Monte Carlo runs.

Figure 4.9 shows the Monte Carlo average of the OSPA distance for $p = 1$ and $c = 100m$. The ERR sensor management presents a behavior quite similar to the PENT approach. After the initial settle-in phase the curves stabilize to an average error close to $50m$ per target for the PENT and the ERR approaches, and close to $60m$ per target for the random schemes.

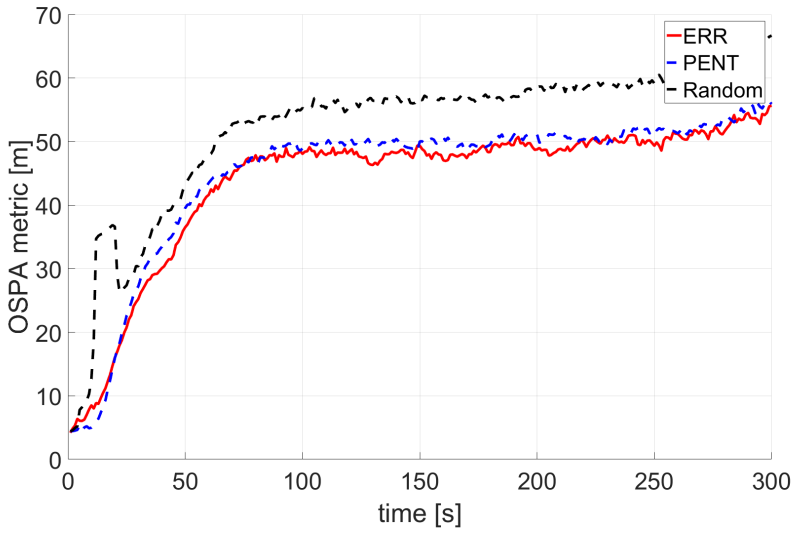

Figure 4.9: OSPA metric. Error performance of the three sensor management approaches without target prioritization, averaged over 1000 Monte Carlo runs.

Examining the cardinality estimation and the localization error given in Figures 4.10 and 4.11, it can be observed that the random schedule achieves an unexpectedly lower localization error, with a similar result to example of binary classification. As far as the localization error is concerned, the results indicate that the ERR and the PENT approaches are penalized much more than the random scheme since they are more successful at keeping track of more targets at the expense of the localization estimation, which is less accurate. Even though the PENT and the PENTI approaches turned out to be effective in this case study, they are not expected to perform well when confronted with target localization problems [RVC11; Del12]. To get a deeper insight into pros and cons of each method, the performance analyses regarding other scenarios are required.

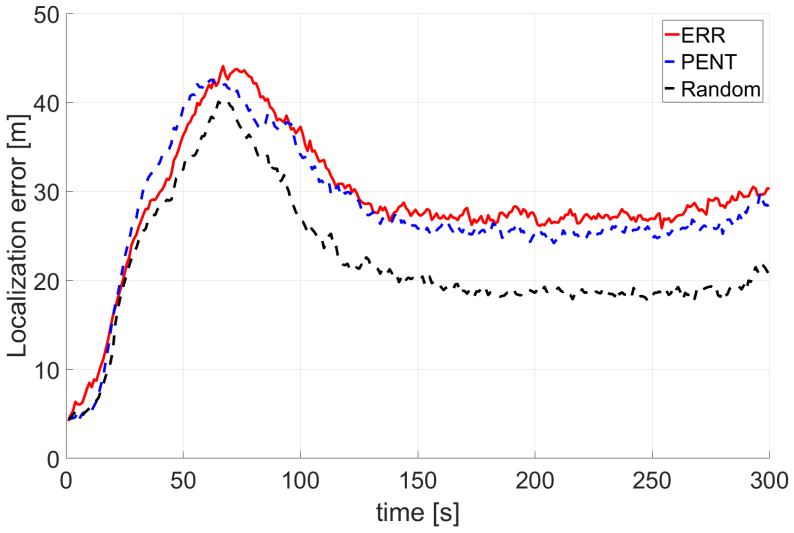

Figure 4.10: Localization error performance of the three sensor management approaches without target prioritization, averaged over 1000 Monte Carlo runs.

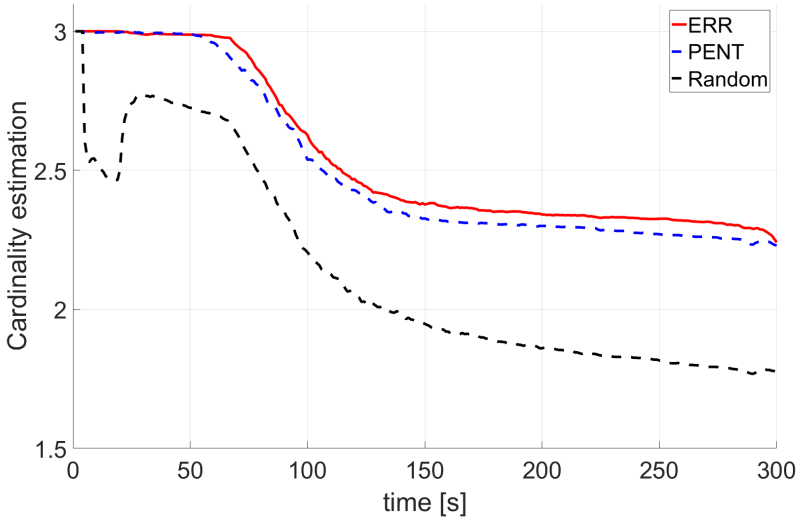

Figure 4.11: Estimated number of targets for three sensor management approaches without target prioritization, averaged over 1000 Monte Carlo runs.

4.5 Conclusion

This chapter introduces the Expected Risk Reduction (ERR) approach to sensor management in the case of a radar beam-pointing problem by combining POMDP theory and the RFS framework. The ERR is based on the expected cost of an incorrect decision about a target's classification. This cost is then conditioned on the event of losing a target track, which allows for achieving the combination of a classification and kinematic uncertainty in the same metric. It has been suggested that the ERR approach can maintain a track on targets of interest when it is not possible for a sensor to track all targets in the environment. The GM-PHD tracker was employed to efficiently but approximately propagate the mutli-object posterior density, which was then used to calculate a reward function in order to determine the sensor control. The numerical examples demonstrated the effectiveness of the proposed reward function for PHD filtering in a radar beam-pointing problem where targets need to be prioritized.

Conclusion and future work

This dissertation presented new methods of estimation and sensor control for tracking an unknown time-varying number of objects using the Labeled Random Finite Set (LRFS) formalism and the Partially Observable Markov Decision Process (POMDP) framework. Two new multi-object tracking methods were proposed, the mono- and multi-sensor Labeled Probability Hypothesis Density (LPHD) filter. The former can be considered an approximation of the δ-GLMB filter, which approximates the multi-object posterior using an LMB RFS in order to simplify the prediction and update steps. This approximation results in an accurate and real-time target tracking algorithm. The second proposed algorithm derives a tractable multi-sensor multi-object Bayes filter within the labeled RFS and superpositional sensors. Lastly, the problem of sensor control was addressed with the use of newly derived analytical expressions for the Expected Risk Reduction (ERR) metric between two densities.

The Labeled Random Finite Set (LRFS) provides a theoretical foundation for Bayes-optimal multi-target tracking. A well-known method based on this approach is the Generalized Labeled Multi-Bernoulli (GLMB) filter, the first computationally tractable and provably Bayes-optimal multi-target tracker. Using this as a base, this thesis explored the relationship between the Labeled Multi-Bernoulli (LMB) filter, an approximation of the GLMB filter that uses the PHD density to represent its posterior density, and an analogous labeled version of the PHD filter. As a result, we presented the rigorous theoretical formulation of the mono-sensor Labeled Probability Hypothesis Density (LPHD) filter with its Gaussian-mixture implementation.

In contrast to the heuristic solutions to the PHD filter with labels, the proposed LPHD filter completely implements the labeled RFS theory, which facilitates the representation of labeled objects and presents a more accurate approximation of the multi-object Bayes filter. However, the computational complexity of the LPHD filter update is significantly higher than other implementations of the PHD filter using the standard RFS. Additionally, the LPHD filter facilitates an implementation using Gaussian mixtures and does not tend to degrade when the clutter is dense or the objects are close.

The second contribution of this thesis is the extension of the mono-sensor LPHD filter for superpositional sensors, resulting in the theoretical formulation of the multi-sensor LPHD filter. The multi-object Bayes-optimal filter for non-standard measurement models is still computationally intractable. Numerous efforts were invested by Mahler, Nannuru, Papi, Saucan et al. to propose a computationally tractable multi-sensor algorithm within the standard and labeled RFS. Thus, as a continuation of their work, we derive a fast multi-sensor multi-object Bayes filter within the labeled RFS that can computationally tractable. The computational cost reduction can be expected to be achieved at the expense of object state estimation performance. The proposed derivation is also applicable to approximations such as the Labeled Multi-Bernoulli (LMB) filter since this filter requires a special case of the GLMB RFS.

Finally, the Expected Risk Reduction (ERR) sensor management method based on the minimization of the Bayes risk was introduced. The ERR approach is derived from the expected cost of an incorrect decision about a target's classification. This cost is then conditioned on the event of losing a target track, which allows for the combination of a classification and kinematic uncertainty in the same metric. We have suggested that the ERR approach can maintain a track on targets of interest when it is not possible for a sensor to track all targets in the environment. The GM-PHD tracker was employed to efficiently but inexactly propagate the multi-target posterior density, which was then used to calculate a reward function in order to determine the sensor control. The numerical examples demonstrated the effectiveness of the proposed reward function for PHD filtering in a radar beam-pointing problem where targets need to be prioritized.

The algorithms developed in this thesis can be used for practical tracking applications and compare well with conventional algorithms such as the Joint Probabilistic Data Association filter (JPDAF) and the Multiple Hypothesis Tracking (MHT). We focused on real-time solutions, and even the algorithms based on the RFS framework avoid explicit data association, it is still demanding many approximations to be computationally tractable and able to turn in quasi-real-time. The difference between the conventional algorithms and those based on the RFS framework is more connected to the rigorous mathematical formalism than to the computational performance. However, the mathematical rigor used in the RFS formalism allows derivations and approximations for general cases, whereas solutions based on heuristics are limited to specific solutions. Much effort will still be needed to achieve real-time algorithms for sensor control and multi-object tracking within the standard and labeled RFS framework. However, the advances in accelerated computing will make it possible by using computational parallelization and possibly a new mathematical formalism.

Future work

The following are some potential areas for future research in multi-object estimation and control based on this thesis:

- Develop a second-order LPHD using the idea proposed by Schlangen et al. [Sch+18]. Due to the LMB RFS properties, it is possible to calculate the variance of the number of targets in a simple manner, which can facilitate the second-order LPHD implementation and integration with sensor management platforms.
- Explore different methods for gating, pruning, and truncation techniques that can be implemented to reduce the complexity of mono- and the multi-sensor LPHD filters.
- Development of the Sequential Monte Carlo (SMC) version of the LPHD filter, as well as support of the multi-model approach, simulations applicable to Simultaneous Localization and Mapping (SLAM), and different birth models.
- Develop and implement a GM and SMC version of the multi-sensor LPHD filter. This could potentially be based on works of Papi and Beard et al. [Pap+14; BVV15].
- Investigate computationally efficient solutions using the ERR metric for the LPHD filter with multiple-steps-ahead sensor control as well as sensor management in distributed fusion architecture.

Appendix A

Finite Set Statistics

This appendix presents the rules necessary to use Finite Set Statistics (FISST). They are summarized below for reference. For details, see Mahler [Mah07b; Mah14].

A.1 Sum rule

$$\frac{\delta}{\delta Z}\Big[a_1\beta_1(S) + a_2\beta_2(S)\Big] = a_1\frac{\delta\beta_1}{\delta Z}(S) + a_2\frac{\delta\beta_2}{\delta Z}(S) \tag{A.1}$$

$$\int\Big[a_1F_1(S) + a_2F_2(S)\Big]\delta Z = a_1\int F_1(S)\delta Z + a_2\int F_2(S)\delta Z \tag{A.2}$$

A.2 Product rule

$$\frac{\delta}{\delta z}\Big[\beta_1(S)\beta_2(S)\Big] = \frac{\delta\beta_1}{\delta z}(S)\beta_2(S) + \frac{\delta\beta_2}{\delta z}(S)\beta_1(S) \tag{A.3}$$

$$\frac{\delta}{\delta Z}\Big[\beta_1(S)\beta_2(S)\Big] = \sum_{W\subseteq Z}\frac{\delta\beta_1}{\delta W}(S)\frac{\delta\beta_2}{\delta(Z-W)}(S) \tag{A.4}$$

A.3 Constant rule

$$\frac{\delta}{\delta Z}K = 0 \tag{A.5}$$

A.4 Chain rule

$$\frac{\delta}{\delta z} f(\beta(S)) = \frac{\mathrm{d}f}{\mathrm{d}x}(\beta(S)) \frac{\delta \beta}{\delta z}(S) \tag{A.6}$$

$$\frac{\delta}{\delta z} f(\beta_1(S) \cdots \beta_n(S)) = \sum_{i=1}^{n} \frac{\partial f}{\partial x_i} \Big(\beta_1(S) \cdots \beta_n(S) \Big) \frac{\delta \beta_i}{\delta z}(S) \tag{A.7}$$

A.5 Power rule

Let $p(S)$ be a probability mass function with density function $f_p(z)$, where $Z = \{z_1, \cdots, z_k\}$. Then:

$$\frac{\delta}{\delta Z} p(S)^n = \begin{cases} \dfrac{n!}{(n-k)!} p(S)^{n-k} f_p(z_k) & \text{if } k \leq n \\ 0 & \text{if } k > n \end{cases} \tag{A.8}$$

A.6 Random set model for target dynamic and sensors

Random set notation for target motion model is defined by:

$$\Gamma_{k+1} = \Phi_k(X_k, V_k) \cup B_k(X_k) \tag{A.9}$$

where Φ represents the change of target dynamics from time k to $k+1$. B caters to the target birth process in the multiple target case. Similarly, the sensor model in random set notation is defined by:

$$\Sigma = T(X) \cup C(X) \tag{A.10}$$

where T defines the measurements that originated from true targets, while C accounts for clutter measurements.

A.7 Belief-mass function of sensor model

The probability mass $p(S|x) = P(Z \in S)$ captures the statistical behavior of observation set Z. In random set domain, the statistics of Σ are characterized by its belief-mass function $\beta(S|X)$. The belief-mass measure is defined by:

$$\beta(S|X) = \beta_{\Sigma|\Gamma}(S|X) = P(\Sigma \subseteq S) \tag{A.11}$$

This belief-mass measure is the total probability that all observations in a sensor scan will be found in any region S.

A.8 Belief-mass function of target model

The probability mass $p(S|x_k) = P(X_{k+1} \in S)$ gives the probability that the target state $X_{k=1}$ will be found in the region S conditioned on previous state x_k. Thus, the Markov transition density can be calculated from the belief-mass measure as follows:

$$\beta_{\Gamma_{k+1}} = P(\Gamma_{k+1} \subseteq S) \tag{A.12}$$

The belief-mass measure of a certain random finite set is given by the following equation:

$$\beta(S|B) = P(A \in S|B) \int_S f(A|B)\delta A \tag{A.13}$$

where A denotes either the target dynamic set Γ_k or the sensor observation set Σ. The integration refers to the sum of densities or likelihood of all possibilities suggested by the random set.

A.9 Set integral

Assuming a function $F(Y)$ is given for a finite set variable Y, the function $F(Y)$ can be represented by the following forms:

$$\begin{aligned} F(\emptyset) &= \text{probability that } Y = \emptyset \\ F(\{y\}) &= \text{likelihood that } Y = \{y\} \\ F(\{y_1, y_2\}) &= \text{likelihood that } Y = \{y_1, y_2\} \\ &\vdots \\ F(\{y_1, \cdots, y_j\}) &= \text{likelihood that } Y = \{y_1, \cdots, y_j\} \end{aligned} \tag{A.14}$$

In general, $F(Y)$ can be a likelihood $F(Z) = f(Z|X)$ or a Markov density $F(X) = f_{k+1}(X|X_k)$. The random finite set integral is defined by:

$$\begin{aligned} \int F(Y)\delta Y &= \sum_{i=0}^{\infty} \frac{1}{i!} \int F(\{y_1, \cdots, y_i\}) \mathrm{d}y_1 \cdots \mathrm{d}y_i \\ &= F(\emptyset) + \sum_{i=1}^{\infty} \frac{1}{i!} \int F(\{y_1, \cdots, y_i\}) \mathrm{d}y_1 \cdots \mathrm{d}y_i \\ &\stackrel{\text{def}}{=} F(\emptyset) + \sum_{i=1}^{\infty} \int F_i(y_1, \cdots, y_i) \mathrm{d}y_1 \cdots \mathrm{d}y_i \end{aligned} \tag{A.15}$$

where $F_i(\cdot)$ is defined by:

$$F_i(y_1, \cdots, y_i) = \begin{cases} \frac{1}{i!} F(\{y_1, \cdots, y_i\}) & \text{if } |\{y_1, \cdots, y_i\}| = i \\ 0 & \text{otherwise} \end{cases} \tag{A.16}$$

A.10 Set derivative

For a random finite set $Y = \{y_1, \cdots, y_m\}$, the set derivatives are defined as follows:

$$\frac{\delta \beta}{\delta y_j}(S) = \frac{\delta}{\delta y_j}\beta(S) = \lim_{\lambda E_y} \frac{\beta(S \cup E_y) - \beta(S)}{\lambda(E_z)} \tag{A.17}$$

$$\frac{\delta \beta}{\delta Y}(S) = \frac{\delta^m}{\partial y_1 \cdots \delta y_m}\beta(S) = \frac{\delta}{\delta y_1} \cdots \frac{\delta}{\delta y_m}\beta(S) \tag{A.18}$$

$$\frac{\delta \beta}{\delta \phi}(S) = \beta(S) \tag{A.19}$$

A.11 Likelihood and Markov densities

According to the random finite set derivative and integral rules, the belief-mass measure and densities are related to each other as follows:

$$\beta(S) = \int_S \frac{\delta \beta}{\delta X}(\phi)\delta X \tag{A.20}$$

$$F(X) = \left[\frac{\delta}{\delta X}\int_S F(Y)\delta Y\right]_{S=\phi} \tag{A.21}$$

Based on the target dynamic model and sensor model, the densities can be calculated after constructing the belief-mass measure of the appropriate sets. After the likelihood and Markov density are obtained, the Bayesian recursion updates the target state in the usual manner.

The true likelihood $f(Z|X)$ is given by:

$$f(Z|X) = \frac{\delta \beta}{\delta Z}(\phi|X) \tag{A.22}$$

The true Markov density is given by:

$$f_{k+1|k}(X_{k+1}|X_k) = \frac{\delta \beta_{k+1|k}}{\delta X_{k+1}}(\phi|X_k) \tag{A.23}$$

Appendix B

Gaussian identities

The derivation of the Gaussian-Mixture Labeled PHD (GM-LPHD) filter often uses the following Gaussian identities introduced by Ho and Lee, which have been used to derive the Kalman filter in the Bayesian form [HL64].

Lemma B.1. *Considering a Gaussian process model with system matrix F and covariance matrix Q:*

$$f(x|\zeta) \sim \mathcal{N}\left(x; F\zeta, Q\right)$$

and a Gaussian distribution with mean $\hat{x}$ and covariance P given by:

$$p(x|\zeta) \sim \mathcal{N}\left(\zeta; \hat{x}, P\right)$$

then the prediction of the Gaussian distribution $p(x|\zeta)$ is defined by:

$$\int \mathcal{N}\left(x; F\zeta, Q\right) \mathcal{N}\left(\zeta; \hat{x}, P\right) \mathrm{d}\zeta = \mathcal{N}\left(x; F\hat{x}, FPF^T\right) \tag{B.1}$$

Lemma B.2. *Considering a linear Gaussian measurement model with observation matrix H and observation noise covariance R given by:*

$$g(z|x) \sim \mathcal{N}\left(z; Hx, R\right)$$

and a predicted Gaussian distribution state with mean $\hat{x}_+$ and covariance matrix P_+ represented by:

$$p_+(x) \sim \mathcal{N}\left(x; \hat{x}_+, P_+\right)$$

then the innovation of the Gaussian distribution $p_+(x)$ is defined by:

$$\mathcal{N}(z; Hx, R)\ \mathcal{N}\left(\zeta; \hat{x}, P\right)\ \mathcal{N}\left(x; \hat{x}_+, P_+\right) = q(z)\ \mathcal{N}\left(x; \hat{x}, P\right) \tag{B.2}$$

where:

$$\begin{aligned} q(z) &= \mathcal{N}\left(z; H\hat{x}_+, HP_+H^T + R\right) \\ \hat{x} &= \hat{x}_+ + K(z - H\hat{x}_+) \\ P &= P_+ - KSK^T \\ S &= HP_+H^T + R \\ K &= P_+H^TS^{-1} \end{aligned}$$

Appendix C

Kalman filter derivation

Our goal is to estimate the state x of a stochastic system within a process of the form:

$$x_k = F_{k-1}x_{k-1} + v_{k-1} \tag{C.1}$$

It is assumed that the measurement is linearly related to the state by an equation of the form:

$$z_k = H_k x_k + w_k \tag{C.2}$$

where

$x \in \mathbb{R}^n$ is the state vector of the process at time k

$z \in \mathbb{R}^\ell$ is the measurement vector of x at time k

$F \in \mathbb{R}^{n\times n}$ is the transition matrix of the process

$H \in \mathbb{R}^{\ell\times n}$ is the observation model matrix used to map the state x into the measurement domain

$v \in \mathbb{R}^r$ is the process noise vector

$w \in \mathbb{R}^\ell$ is the measurement noise vector

Accordingly, it suffices to look for an updated estimate $x_{k|k}$ that is a linear function of *a priori* estimate and the measurement z_k:

$$x_{k|k} = \overline{K}_k x_{k|k-1} + K_k z_k \tag{C.3}$$

where $x_{k|k-1}$ and $x_{k|k}$ are the *a priori* and *a posteriori* estimate of x_k. The matrices $\overline{K}_k$ and K_k are unknown. We seek those values of $\overline{K}_k$ and K_k such that the new estimate $x_{k|k}$ will satisfy the following orthogonality condition:

$$\mathbb{E}[(x_k - x_{k|k})z_i^T] = 0, \qquad i = 1, 2, \cdots, k-1 \tag{C.4}$$
$$\mathbb{E}[(x_k - x_{k|k})z_k^T] = 0 \tag{C.5}$$

Substituting (C.3) into (C.4), the expression becomes:

$$\mathbb{E}[(x_k - \overline{K}_k x_{k|k-1} - K_k z_k)z_i^T] = 0 \tag{C.6}$$

The measurement z_k is given by (C.2). Then the above expression can be rewritten as:

$$\mathbb{E}[(x_k - \overline{K}_k x_{k|k-1} - K_k H_k x_k - K_k w_k)z_i^T] = 0 \tag{C.7}$$

It is possible to rewritte the above equation subtracting and adding $\overline{K}_k x_k$:

$$\begin{aligned} 0 &= \mathbb{E}[(x_k - \overline{K}_k x_k - \overline{K}_k x_{k|k-1} + \overline{K}_k x_k - K_k H_k x_k - K_k w_k)z_i^T] \\ &= \mathbb{E}[(x_k - \overline{K}_k x_k - \overline{K}_k (x_{k|k-1} - x_k) - K_k H_k x_k - K_k w_k)z_i^T] \\ &= \mathbb{E}[x_k z_i^T] - \overline{K}_k \,\mathbb{E}[x_k z_i^T] - \overline{K}_k \,\mathbb{E}[(x_{k|k-1} - x_k)z_i^T] - K_k H_k \,\mathbb{E}[x_k z_i^T] - K_k \,\mathbb{E}[w_k z_i^T] \end{aligned} \tag{C.8}$$

The term $\mathbb{E}[(x_{k|k-1} - x_k)z_i^T] = 0$ due to the orthogonality condition. The measurement noise w_k is assumed uncorrelated with the measurement z_i, i.e., $\mathbb{E}[w_k z_i^T] = 0$ for $i = 1, \cdots, k-1$. Thus, the equation can be reduced to the form:

$$\begin{aligned} 0 &= \mathbb{E}[x_k z_i^T] - \overline{K}_k \,\mathbb{E}[x_k z_i^T] - K_k H_k \,\mathbb{E}[x_k z_i^T] \\ &= (I - \overline{K}_k - K_k H_k) \,\mathbb{E}[x_k z_i^T] \end{aligned} \tag{C.9}$$

The above equation can be satisfied for any given x_k if:

$$\overline{K}_k = I - K_k H_k \tag{C.10}$$

The errors can be calculated as follows:

$$\tilde{x}_{k|k} \stackrel{\text{def}}{=} x_{k|k} - x_k \tag{C.11}$$
$$\tilde{x}_{k|k-1} \stackrel{\text{def}}{=} x_{k|k-1} - x_k \tag{C.12}$$

$$\tilde{z}_k \stackrel{\text{def}}{=} z_{k|k-1} - z_k = H_k x_{k|k-1} - z_k \tag{C.13}$$

where $\tilde{x}_{k|k-1}$ and $\tilde{x}_{k|k}$ are the estimation errors before and after updates. The estimation $x_{k|k}$ depends linearly on x_k, which depends linearly on z_k. Accordingly, from (C.5):

$$\mathbb{E}[(x_k - x_{k|k})z_{k|k-1}^T] = 0 \tag{C.14}$$

and subtracting (C.5) from the above equation:

$$\mathbb{E}[(x_k - x_{k|k})\tilde{z}_k^T] = 0 \tag{C.15}$$

Substituting $x_{k|k}$ and $\tilde{z}_k$ from (C.3) and (C.13). Then:

$$\mathbb{E}[(x_k - \overline{K}_k x_{k|k-1} - K_k z_k)(H_k x_{k|k-1} - z_k)^T] = 0 \tag{C.16}$$

Replacing $\overline{K}_k$, z_k, $\tilde{x}_{k|k}$, the above equation que be modified as follows:

$$\begin{aligned}
0 &= \mathbb{E}[(x_k - x_{k|k-1} + K_k H_k x_{k|k-1} - K_k H_k x_k - K_k w_k)(H_k x_{k|k-1} - H_k x_k - w_k)^T] \\
&= \mathbb{E}[((x_k - x_{k|k-1}) + K_k H_k (x_{k|k-1} - x_k) - K_k w_k)(H_k (x_{k|k-1} - x_k) - w_k)^T] \\
&= \mathbb{E}[(-\tilde{x}_{k|k-1} + K_k H_k \tilde{x}_{k|k-1} - K_k w_k)(H_k \tilde{x}_{k|k-1} - w_k)^T]
\end{aligned} \tag{C.17}$$

Since $\mathbb{E}[\tilde{x}_{k|k-1} w_k^T] = 0$, and defining $\mathbb{E}[w_k w_k^T]$ to be the measurement noise covariance R_k, and $P_{k|k-1}$ to be the *a priori* covariance $P_{k|k-1} \overset{\text{def}}{=} \mathbb{E}[\tilde{x}_{k|k-1} \tilde{x}_{k|k-1}^T]$, then the above equation becomes:

$$\begin{aligned}
0 =& \mathbb{E}[-\tilde{x}_{k|k-1}\tilde{x}_{k|k-1}^T H_k^T + \tilde{x}_{k|k-1} w_k^T + K_k H_k \tilde{x}_{k|k-1}\tilde{x}_{k|k-1}^T H_k^T \\
&- K_k H_k \tilde{x}_{k|k-1} w_k^T - K_k w_k \tilde{x}_{k|k-1}^T H_k^T + K_k w_k w_k^T] \\
=& - \mathbb{E}[\tilde{x}_{k|k-1}\tilde{x}_{k|k-1}^T] H_k^T + \mathbb{E}[\tilde{x}_{k|k-1} w_k^T] + K_k H_k \, \mathbb{E}[\tilde{x}_{k|k-1}\tilde{x}_{k|k-1}^T] H_k^T \\
&- K_k H_k \, \mathbb{E}[\tilde{x}_{k|k-1} w_k^T] - K_k \, \mathbb{E}[w_k \tilde{x}_{k|k-1}^T] H_k^T + K_k \, \mathbb{E}[w_k w_k^T] \\
=& - P_{k|k-1} H_k^T + K_k H_k P_{k|k-1} H_k^T + K_k R_k \\
=& - P_{k|k-1} H_k^T + K_k (H_k P_{k|k-1} H_k^T + R_k)
\end{aligned} \tag{C.18}$$

in this way, the Kalman gain can be expressed as:

$$K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1} \tag{C.19}$$

Using the expressions derived for $\overline{K}_k$ and K_k, it is possible to derive an expression for the *a posteriori* covariance error, which is defined in a similar manner to the *a priori* covariance error $P_{k|k-1}$:

$$P_{k|k} = \mathbb{E}[\tilde{x}_{k|k} \tilde{x}_{k|k}^T] \tag{C.20}$$

Substituting (C.10) into (C.3):

$$\begin{aligned} x_{k|k} &= \overline{K}_k x_{k|k-1} + K_k z_k \\ &= (I - K_k H_k) x_{k|k-1} + K_k z_k \end{aligned} \tag{C.21}$$

Subtracting x_k from both sides of the above equation and replacing z_k from (C.2), we have:

$$\begin{aligned} x_{k|k} - x_k &= -x_k + (I - K_k H_k) x_{k|k-1} + K_k H_k x_k + K_k w_k \\ x_{k|k} - x_k &= -x_k + x_{k|k-1} - K_k H_k x_{k|k-1} + K_k H_k x_k + K_k w_k \\ x_{k|k} - x_k &= -x_k + x_{k|k-1} - K_k H_k (x_{k|k-1} - x_k) + K_k w_k \\ \tilde{x}_{k|k} &= \tilde{x}_{k|k-1} - K_k H_k \tilde{x}_{k|k-1} + K_k w_k \\ \tilde{x}_{k|k} &= (I - K_k H_k) \tilde{x}_{k|k-1} + K_k w_k \end{aligned} \tag{C.22}$$

Replacing the above equation into (C.20) and considering that $\mathbb{E}[\tilde{x}_{k|k-1} w_k^T] = 0$, we have:

$$\begin{aligned} P_{k|k} =& \mathbb{E}[\tilde{x}_{k|k} \tilde{x}_{k|k}^T] \\ =& \mathbb{E}[((I - K_k H_k)\tilde{x}_{k|k-1} + K_k w_k)((I - K_k H_k)\tilde{x}_{k|k-1} + K_k w_k)^T] \\ =& \mathbb{E}[(I - K_k H_k)\tilde{x}_{k|k-1}(I - K_k H_k)^T \tilde{x}_{k|k-1}^T + K_k w_k w_k^T K_k^T] \\ =& (I - K_k H_k)\, \mathbb{E}[\tilde{x}_{k|k-1} \tilde{x}_{k|k-1}^T](I - K_k H_k)^T + K_k\, \mathbb{E}[w_k w_k^T] K_k^T \\ =& (I - K_k H_k) P_{k|k-1} (I - K_k H_k)^T + K_k R K_k^T \end{aligned} \tag{C.23}$$

This last equation is known as "Joseph form" of the covariance update equation [BJ05]. This equation can be rewritten replacing K_k from (C.19):

$$\begin{aligned} P_{k|k} =& P_{k|k-1} - P_{k|k-1} H_k^T K_k^T - K_k H_k P_{k|k-1} + K_k H_k P_{k|k-1} H_k^T K_k^T + K_k R K_k^T \\ =& (I - K_k H_k) P_{k|k-1} - P_{k|k-1} H_k^T K_k^T + \underbrace{K_k (H_k P_{k|k-1} H_k^T + R)}_{P_{k|k-1} H_k^T} K_k^T \\ =& (I - K_k H_k) P_{k|k-1} \end{aligned} \tag{C.24}$$

This form is the one most often used in computation. The *a priori* covariance $P_{k|k-1}$ can be expressed considering the *a priori* state estimate:

$$x_{k|k-1} = F_{k-1} x_{k-1|k-1} \tag{C.25}$$

subtracting x_k from both sides of the above equation, then using (C.1) we have:

$$
\begin{aligned}
x_{k|k-1} - x_k &= F_{k-1}x_{k-1|k-1} - x_k \\
\tilde{x}_{k|k-1} &= F_{k-1}x_{k-1|k-1} - x_k \\
&= F_{k-1}x_{k-1|k-1} - F_{k-1}x_{k-1} - v_{k-1} \\
&= F_{k-1}(x_{k-1|k-1} - x_{k-1}) - v_{k-1} \\
&= F_{k-1}\tilde{x}_{k-1|k-1} - v_{k-1}
\end{aligned} \tag{C.26}
$$

using the above equation and defing $\mathbb{E}[v_k v_k^T]$ as the process noise covariance Q_k:

$$
\begin{aligned}
P_{k|k-1} &\stackrel{\text{def}}{=} \mathbb{E}[\tilde{x}_{k|k-1}\tilde{x}_{k|k-1}^T] \\
&= \mathbb{E}[(F_{k-1}\tilde{x}_{k-1|k-1} - v_{k-1})(F_{k-1}\tilde{x}_{k-1|k-1} - v_{k-1})^T] \\
&= F_{k-1}\,\mathbb{E}[\tilde{x}_{k-1|k-1}\tilde{x}_{k-1|k-1}^T]F_{k-1}^T - F_{k-1}\,\mathbb{E}[\tilde{x}_{k-1|k-1}v_{k-1}^T] \\
&\quad - \mathbb{E}[v_{k-1}\tilde{x}_{k-1|k-1}^T]F_{k-1}^T + \mathbb{E}[v_{k-1}v_{k-1}^T] \\
&= F_{k-1}P_{k-1|k-1}F_{k-1}^T + Q_k
\end{aligned} \tag{C.27}
$$

which gives the *a priori* covariance matrix of estimation uncertainty as a function of the previous *a posteriori* covariance matrix.

Appendix D

Kinematic models for Target Tracking

The key to successful target tracking lies in the effective extraction of the useful information about the target's state. The information extraction is performed using models to describe the target dynamics and sensor characteristics. Most of the target tracking algorithms base their performance on *a priori* knowledge of the target's kinematic and the sensor system which is assumed to be sufficiently accurate.

The primary objective of target tracking is to estimate the state trajectories of a moving object. Almost all maneuvering target tracking algorithms are model-based. They assume that the target motion and its observations can be represented by some known mathematical models in a precise way. In general, the target is modeled as a punctual mass and its dynamic described by a linear state-space model. A target dynamic model describes the evolution of the target state with respect to time. The dynamic model of a target can be described by discrete- or continuous-time models [BP99; BLK01; Cha+11; BWT11]. This thesis only considers the discrete-time models. Hence, the target state x_k evolves in time according to the state transition equation:

$$\begin{cases} x_k = \mathrm{f}_{k-1}(x_{k-1}, v_{k-1}) \\ z_k = h_k(x_k, w_k) \end{cases} \tag{D.1}$$

where f_{k-1} is a known, possible nonlinear function that transforms any given state vector x_{k-1} and process noise v_{k-1} at time $k-1$ into a new state vector x_k at time k. The function h_k is a known, possibly nonlinear function that transform any given state vector x_k and observation noise w_k at time k into an observation vector z_k. Note that the process and observation noise are assumed to be uncorrelated. The discrete-time model described in (D.1) is often obtained by discretizing the following continuous-time model [BLK01; LJ03; BWT11; Cha+11]:

$$\begin{aligned} \dot{x}(t) &= A(t)x(t) + D(t)\tilde{v} \\ z(t) &= C(t)x(t) + \tilde{w}(t) \end{aligned} \tag{D.2}$$

where

$x \in \mathbb{R}^n$ is the state vector

$z \in \mathbb{R}^m$ is the measurement vector

$A \in \mathbb{R}^{n\times n}$ is the transition matrix of the dynamic model

$\tilde{v} \in \mathbb{R}^{p}$ is the process noise vector

$D \in \mathbb{R}^{n\times p}$ is the process noise gain

$C \in \mathbb{R}^{m\times n}$ is the measurement model matrix

$\tilde{w} \in \mathbb{R}^{m}$ is the measurement noise

One of the major challenges for target tracking arises from the target motion uncertainty. This uncertainty refers to the fact that an accurate dynamic model of the target being tracked is not available to the tracker. Target motion modeling is thus one of the first tasks for maneuvering target tracking. It aims at developing a tractable model that accounts well for the effect of target motion.

Constant-velocity model

It is well known that a point moving in 3D can be described by its 3D position and velocity vectors. For instance, $x = [x, \dot{x}, y, \dot{y}, z, \dot{z}]$ can be used as a state vector of such a point in the Cartesian coordinate system, where (x,y,z) are the position coordinates along x, y, and z axes, respectively, and $[\dot{x}, \dot{y}, \dot{z}]$ is the is the velocity vector. When a target is treated as a point object, the nonmaneuvering motion is thus described by the vector-valued equation $\dot{x}(t) = 0$, where $x = [\dot{x}, \dot{y}, z]$. Note that z direction is treated differently because a nonmaneuvering motion is assumed in the horizontal $x - y$ plane.

In practice, this ideal equation is usually modified as $\dot{x}(t) = v(t) \approx 0$, where $w(t)$ is white noise with a "small" effect on x that accounts for unpredictable modeling errors. The corresponding state-space model is given by, with state vector $x = [x, \dot{x}, y, \dot{y}, z]^T$:

$$\dot{x}(t) = \overbrace{\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}}^{A} \overbrace{\begin{bmatrix} x \\ \dot{x} \\ y \\ \dot{y} \\ z \end{bmatrix}}^{x(t)} + \overbrace{\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}^{D} \overbrace{\begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}}^{v(t)} \tag{D.3}$$

The direct discrete-time counterpart of the above continuous-time model is [BLK01; LJ03]:

$$x_{k+1} = \overbrace{\begin{bmatrix} 1 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & T & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}}^{F} \overbrace{\begin{bmatrix} x \\ \dot{x} \\ y \\ \dot{y} \\ z \end{bmatrix}}^{x_k} + \overbrace{\begin{bmatrix} \frac{T^2}{2} & 0 & 0 \\ T & 0 & 0 \\ 0 & \frac{T^2}{2} & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{bmatrix}}^{G} \overbrace{\begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}}^{v_k} \tag{D.4}$$

where T is the sampling interval. Note that v_k and v_y correspond to noisy accelerations along x and y axes, while v_z corresponds to noisy velocity along z axis. If w is uncoupled across its components, then the nonmaneuvering motion modeled by the above models is uncoupled

across x, y, and z directions. In this case, the covariance of the noise term in (D.4) is given by:

$$\operatorname{cov}(Gw_k) = \begin{bmatrix} \operatorname{var}(v_x)\frac{T^4}{4} & \operatorname{var}(v_x)\frac{T^3}{2} & 0 & 0 & 0 & 0 \\ \operatorname{var}(v_x)\frac{T^3}{2} & \operatorname{var}(v_x)T^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \operatorname{var}(v_y)\frac{T^4}{4} & \operatorname{var}(v_y)\frac{T^3}{2} & 0 & 0 \\ 0 & 0 & \operatorname{var}(v_y)\frac{T^3}{2} & \operatorname{var}(v_y)T^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \operatorname{var}(v_z)\frac{T^4}{4} & \operatorname{var}(v_z)\frac{T^3}{2} \\ 0 & 0 & 0 & 0 & \operatorname{var}(v_z)\frac{T^3}{2} & \operatorname{var}(v_z)T^2 \end{bmatrix} \tag{D.5}$$

The direct discrete-time equivalent of the above model is [BLK01; LJ03]:

$$x_{k+1} = \overbrace{\begin{bmatrix} 1 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & T & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}}^{F} \overbrace{\begin{bmatrix} x \\ \dot{x} \\ y \\ \dot{y} \\ z \end{bmatrix}}^{x_k} + \overbrace{\begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}}^{v_k} \tag{D.6}$$

where

$$\operatorname{cov}(w_k) = \begin{bmatrix} S_x\frac{T^3}{3} & S_x\frac{T^2}{2} & 0 & 0 & 0 & 0 \\ S_x\frac{T^2}{2} & S_xT & 0 & 0 & 0 & 0 \\ 0 & 0 & S_y\frac{T^3}{3} & S_y\frac{T^2}{2} & 0 & 0 \\ 0 & 0 & S_y\frac{T^2}{2} & S_yT & 0 & 0 \\ 0 & 0 & 0 & 0 & S_z\frac{T^3}{3} & S_z\frac{T^2}{2} \\ 0 & 0 & 0 & 0 & S_z\frac{T^2}{2} & S_zT \end{bmatrix} \tag{D.7}$$

where S is the power spectral density of the discrete-time process noise. The above models (D.3) and (D.4) are known as the continuous- and discrete-time constant-velocity (CV) models.

Bibliography

[Kál60] R. E. Kálmán. "A new Approach to Linear Filtering and Prediction Problems". In: *Transactions of the AMSE–Journal of the Basic Engineering* 82.Series D (1960), pp. 35–45. DOI: 10.1115/1.3662552.

[HL64] Y.-C. Ho and R. C. K. Lee. "A Bayesian approach to problems in stochastic estimation and control". In: *IEEE Transactions on Automatic Control* 9.4 (1964), pp. 333–339.

[Mur68] K. G. Murthy. "An algorithm for ranking all the assignments in order of increasing costs". In: *Operations Research* 16.3 (1968), pp. 682–687.

[Jaz70] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*. Vol. 64. New York: Academic Press, 1970.

[Yen71] J. Y. Yen. "Finding the k shortest loopless paths in a network". In: *Management Science* 17.11 (1971), pp. 712–716.

[Bar74] Y. Bar-Shalom. "Extension of the Probabilistic Data Association Filter to Multitarget Environments". In: *Proceedings of the 5th Symp. Nonlinear Estimation Theory and Its Applications, San Diego, CA* (1974), pp. 16–21.

[Ken74] D. G. Kendall. "Foundations of a theory of random sets". In: *Stochastic Geometry* (1974), pp. 322–376.

[Mat75] G. Matheron. *Random sets and integral geometry*. Wiley New York, 1975.

[Rei79] D. B. Reid. "An algorithm for tracking multiple targets". In: *IEEE transactions on Automatic Control* 24.6 (1979), pp. 843–854.

[Ser80] J. Serra. "The Boolean model and random sets". In: *Computer Graphics and Image Processing* 12.2 (1980), pp. 99–126.

[Shi84] A. N. Shiryaev. *Probability*. New York: Springer-Verlag, 1984.

[DV88] D. J. Daley and D. Vere-Jones. *An introduction to the theory of point processes*. New York: Springer, 1988.

[Hin91] K. J. Hintz. "A measure of the information gain attributable to cueing". In: *IEEE Transactions on Systems, Man, and Cybernetics* 21.2 (1991), pp. 434–442.

[HM91] K. J. Hintz and E. S. McVey. "Multi-process constrained estimation". In: *IEEE Transactions on Systems, Man, and Cybernetics* 21.1 (1991), pp. 237–244.

[MD92] J. M. Manyika and H. F. Durrant-Whyte. "On sensor management in decentralized data fusion". In: *Proceedings of the 31st IEEE Conference on Decision and Control*. Vol. 4. IEEE. 1992, pp. 3506–3507. DOI: 10.1109/CDC.1992.371006.

[GSS93] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. "Novel approach to nonlinear/non-Gaussian Bayesian state estimation". In: *IEE Proceedings F (Radar and Signal Processing)* 140.2 (1993), pp. 107–113.

[Mah94a] R. P. S. Mahler. "Global Integrated Data Fusion". In: *Proc. 7th Nat. Symp. on Sensor Fusion*. Vol. 1. 1994, pp. 187–199.

[Mah94b] R. P. S. Mahler. "Random-set Approach to Data Fusion". In: *SPIE's International Symposium on Optical Engineering and Photonics in Aerospace Sensing*. International Society for Optics and Photonics. 1994, pp. 287–295.

[JUD95] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte. "A new Approach for Filtering Nonlinear Systems". In: *American Control Conference, Proceedings of the 1995*. Vol. 3. June 1995, pp. 1628–1632. DOI: 10.1109/ACC.1995.529783.

[MD95] J. Manyika and H. Durrant-Whyte. *Data Fusion and Sensor Management: A Decentralized Information-Theoretic Approach*. Prentice Hall PTR, 1995.

[SKM96] D. Stoyan, W. S. Kendall, and J. Mecke. *Stochastic Geometry and its Applications*. New York: John Wiley & Sons, 1996. ISBN: 978-0471950998.

[GMN97] R. Goodman, R. P. S. Mahler, and H. T. Nguyen. *Mathematics of Data Fusion*. Springer Science & Business Media, 1997.

[JU97] S. J. Julier and J. K. Uhlmann. "A New Extension of the Kalman Filter to Nonlinear Systems". In: *Proceedings of SPIE AeroSense*. Vol. 3068. 1997, pp. 182–193. DOI: 10.1117/12.280797.

[Kas97] K. Kastella. "Discrimination Gain to Optimize Detection and Classification". In: *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* 27.1 (Jan. 1997), pp. 112–116. ISSN: 1083-4427. DOI: 10.1109/3468.553230.

[Epp98] D. Eppstein. "Finding the k shortest paths". In: *SIAM Journal on Computing* 28.2 (1998), pp. 652–673.

[BP99] S. Blackman and R. Popoli. *Design and Analysis of Modern Tracking Systems*. Norwood, MA: Artech House, 1999, p. 1230. ISBN: 1-58053-006-0.

[ML99] D. Maltese and A. Lucas. "Data fusion: Quite Silent Search Function in Naval Air Defense". In: *Proceedings of SPIE AeroSense*. Vol. 3698. 1999, pp. 36–47. DOI: 10.1117/12.354543.

[DGA00] A. Doucet, S. Godsill, and C. Andrieu. "On sequential Monte Carlo sampling methods for Bayesian filtering". In: *Statistics and Computing* 10.3 (2000), pp. 197–208.

[JUD00] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte. "A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators". In: *IEEE Transactions on Automatic Control* 45.3 (2000), pp. 477–482. DOI: 10.1109/9.847726.

[Mah00] R. P. S. Mahler. *A theoretical foundation for the Stein-Winter" Probability Hypothesis Density (PHD)" multitarget tracking approach*. Tech. rep. Army Research Office Alexandria VA, 2000.

[Wv00] E. A. Wan and R. van der Merwe. "The Unscented Kalman Filter for Nonlinear Estimation". In: *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*. IEEE. 2000, pp. 153–158.

[BLK01] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. New York, NY, USA: John Wiley & Sons, 2001, p. 558. ISBN: 0-471-41655-X.

[DFG01] A. Doucet, N. D. Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. New York: Springer, 2001. ISBN: 978-0-387-95146-1.

[Mah01a] R. P. S. Mahler. *Multitarget moments and their application to multitarget tracking*. Tech. rep. Lockheed Martin Tactical Defense Systems - Eagan ST Paul MN, 2001.

[Mah01b] R. P. S. Mahler. *Random set theory for target tracking and identification*. 2001.

[Aru+02] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking". In: *IEEE Transactions on signal processing* 50.2 (2002), pp. 174–188.

[Jul02] S. J. Julier. "The Scaled Unscented Transformation". In: *American Control Conference, 2002. Proceedings of the 2002*. Vol. 6. IEEE. May 2002, pp. 4555–4559.

[XS02] N. Xiong and P. Svensson. "Multi-sensor management for information fusion: issues and approaches". In: *Information Fusion* 3.2 (2002), pp. 163–186.

[DV03] D. J. Daley and D. Vere-Jones. *An Introduction to the Theory of Point Processes: Volume I: Elementary Theory and Methods*. 2nd ed. New York: Springer Science & Business Media, 2003, p. 471. ISBN: 978-0-387-21564-8.

[KKH03] C. Kreucher, K. Kastella, and A. O. Hero III. "Multi-target Sensor Management Using Alpha-Divergence Measures". In: *Information Processing in Sensor Networks*. Springer. 2003, pp. 209–222.

[LJ03] X. R. Li and V. P. Jilkov. "Survey of Maneuvering Target Tracking. Part I: Dynamic Models". In: *IEEE Transactions on Aerospace and Electronic Systems* 39.4 (Oct. 2003), pp. 1333–1364. ISSN: 0018-9251. DOI: 10.1109/TAES.2003.1261132.

[Mah03a] R. P. S. Mahler. "Multitarget Bayes Filtering via First-Order Multitarget Moments". In: *IEEE Transactions on Aerospace and Electronic Systems* 39.4 (Oct. 2003), pp. 1152–1178. ISSN: 0018-9251. DOI: 10.1109AES.2003.1261119.

[Mah03b] R. P. S. Mahler. "Objective functions for Bayesian control-theoretic sensor management, 1: Multitarget first-moment approximation". In: *Aerospace Conference, 2003. Proceedings. 2003 IEEE*. Vol. 4. IEEE. Mar. 2003, 4_1905–4_1923. DOI: 10.1109/AERO.2003.1235121.

[MW03] J. Moller and R. P. Waagepetersen. *Statistical inference and simulation for spatial point processes*. CRC Press, 2003.

[Sid03] H. Sidenbladh. "Multi-Target Particle Filtering for the Probability Hypothesis Density". In: *Proceedings of the Sixth International Conference of Information Fusion*. Vol. 2. July 2003, pp. 800–806.

[Tim03] a. R. P. S. M. Tim Zajic. "Particle-Systems Implementation of the PHD Multitarget-Tracking Filter". In: *AeroSense 2003*. 2003, pp. 291–300.

[VSD03] B.-N. Vo, S. Singh, and A. Doucet. "Sequential Monte Carlo Implementation of the PHD Filter for Multi-target Tracking". In: *Proceedings of the Sixth International Conference of Information Fusion*. Vol. 2. July 2003, pp. 792–799. DOI: 10.1109/ICIF.2003.177320.

[HM04] J. R. Hoffman and R. P. S. Mahler. "Multitarget miss distance via optimal assignment". In: *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 34.3 (2004), pp. 327–336.

[JU04] S. J. Julier and J. K. Uhlmann. "Unscented Filtering and Nonlinear Estimation". In: *Proceedings of the IEEE* 92.3 (Mar. 2004), pp. 401–422. ISSN: 0018–9219. DOI: 10.1109/JPROC.2003.823141.

[Mah04] R. P. S. Mahler. "Multitarget sensor management of dispersed mobile sensors". In: *Theory and algorithms for cooperative systems*. World Scientific, 2004, pp. 239–310.

[MZ04] R. P. S. Mahler and T. R. Zajic. "Probabilistic objective functions for sensor management". In: *Proc. SPIE* 5429 (2004), pp. 233–244. DOI: 10.1117/12.543530.

[Pan+04] K. Panta, B.-N. Vo, S. Singh, and A. Doucet. "Probability Hypothesis Density Filter versus Multiple Hypothesis Tracking". In: *Proceedings of SPIE*. Vol. 5429. 2004, pp. 284–295. DOI: 10.1117/12.543357.

[RAG04] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman filter: Particle filters for tracking applications*. Artech House, 2004.

[van04] R. van der Merwe. "Sigma-Point Kalman filters for Probabilistic Inference in Dynamic State-Space Models". PhD thesis. Oregon Health & Science University, 2004.

[BJ05] R. S. Bucy and P. D. Joseph. *Filtering for Stochastic Processes with Applications to Guidance*. Vol. 326. Chelsea Publishing, Providence, RI: American Mathematical Society, 2005.

[CB05] D. E. Clark and J. Bell. "Data association for the PHD filter". In: *Intelligent Sensors, Sensor Networks and Information Processing Conference, 2005. Proceedings of the 2005 International Conference on*. IEEE. 2005, pp. 217–222.

[KHK05] C. Kreucher, A. O. Hero III, and K. Kastella. "A Comparison of Task Driven and Information Driven Sensor Management for Target Tracking". In: *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05*. IEEE. 2005, pp. 4004–4009.

[PVS05] K. Panta, B.-N. Vo, and S. Singh. "Improved Probability Hypothesis Density (PHD) Filter for Multitarget Tracking". In: *Third International Conference on Intelligent Sensing and Information Processing,2005. ICISIP 2005.* Dec. 2005, pp. 213–218. DOI: 10.1109/ICISIP.2005.1619438.

[VM05] B.-N. Vo and W.-K. Ma. "A closed-form solution for the Probability Hypothesis Density filter". In: *7th International Conference on Information Fusion*. Vol. 2. July 2005, 8 pp. DOI: 10.1109/ICIF.2005.1591948.

[Cla06] D. E. Clark. "Multiple Target Tracking with the Probability Hypothesis Density Filter". PhD thesis. Edinburgh, United Kingdom: Heriot Watt University, Department of Electrical, Electronic and Computing Engineering, 2006.

[CB06] D. E. Clark and J. Bell. "Convergence Results for the Particle PHD Filter". In: *IEEE Transactions on Signal Processing* 54.7 (2006), pp. 2652–2661.

[CPV06] D. E. Clark, K. Panta, and B.-N. Vo. "The GM-PHD Filter Multiple Target Tracker". In: *9th International Conference on Information Fusion*. July 2006, pp. 1–8. DOI: 10.1109/ICIF.2006.301809.

[Joh+06] A. M. Johansen, S. S. Singh, A. Doucet, and B.-N. Vo. "Convergence of the SMC Implementation of the PHD Filter". In: *Methodology and Computing in Applied Probability* 8.2 (2006), pp. 265–291.

[Ngu06] H. T. Nguyen. *An Introduction to Random Sets*. CRC press, 2006.

[PVC06] K. Panta, B.-N. Vo, and D. E. Clark. "An Efficient Track Management Scheme for the Gaussian-Mixture Probability Hypothesis Density Tracker". In: *4th International Conference on Intelligent Sensing and Information Processing. ICISIP 2006*. Oct. 2006, pp. 230–235. DOI: 10.1109/ICISIP.2006.4286102.

[VM06] B.-N. Vo and W.-K. Ma. "The Gaussian Mixture Probability Hypothesis Density Filter". In: *IEEE Transactions on Signal Processing* 54.11 (Nov. 2006), pp. 4091–4104. ISSN: 1053-587X. DOI: 10.1109SP.2006.881190.

[CVV07] D. E. Clark, B.-T. Vo, and B.-N. Vo. "Gaussian Particle Implementations of Probability Hypothesis Density Filters". In: *Aerospace Conference, 2007 IEEE*. IEEE. 2007, pp. 1–11.

[Mah07a] R. P. S. Mahler. "PHD Filters of Higher Order in Target Number". In: *IEEE Transactions on Aerospace and Electronic Systems* 43.4 (Oct. 2007), pp. 1523–1543. ISSN: 0018-9251. DOI: 10.1109AES.2007.4441756.

[Mah07b] R. P. S. Mahler. *Statistical Multisource-Multitarget Information Fusion*. Artech House, 2007.

[Pan07] K. Panta. "Multi-Target Tracking Using 1st Moment of Random Finite Sets". PhD thesis. Australia: University of Melbourne, Department of Electrical and Electronic Engineering, 2007.

[PR07] D. Papageorgiou and M. Raykin. "A risk-based approach to sensor resource management". In: *Advances in Cooperative Control and Optimization*. Springer, 2007, pp. 129–144.

[Her+08] A. O. Hero III, D. A. Castañón, D. Cochran, and K. Kastella. *Foundations and Applications of Sensor Management*. USA: Springer Science & Business Media, 2008, p. 308. ISBN: 978-0-387-27892-6. DOI: 10.1007/978-0-387-49819-5.

[SVV08] D. Schuhmacher, B.-T. Vo, and B.-N. Vo. "A Consistent Metric for Performance Evaluation of Multi-Object Filters". In: *IEEE Transactions on Signal Processing* 56.8 (2008), pp. 3447–3457.

[Vo08] B.-T. Vo. "Random Finite Sets in Multi-object Filtering". PhD thesis. Australia: University of Western Australia, School of Electrical, Electronic and Computer Engineering, 2008.

[Mah09a] R. P. S. Mahler. "CPHD filters for superpositional sensors". In: *Signal and Data Processing of Small Targets 2009*. Vol. 7445. International Society for Optics and Photonics. 2009, 74450E.

[Mah09b] R. P. S. Mahler. "The multisensor PHD filter: I. General solution via multitarget calculus". In: *Proceedings of SPIE*. Vol. 7336. 2009, 73360E–73360E-12. DOI: 10.1117/12.818024.

[Mah09c] R. P. S. Mahler. "The multisensor PHD filter: II. Erroneous solution via Poisson magic". In: *Signal Processing, Sensor Fusion, and Target Recognition XVIII*. Vol. 7336. International Society for Optics and Photonics. 2009, p. 73360D.

[Del+10] 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, July 2010, p. 64.

[Md10] D. Macagnano and G. T. F. de Abreu. "Multitarget Tracking with the Cubature Kalman Probability Hypothesis Density Filter". In: *Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers*. IEEE. 2010, pp. 1455–1459.

[RV10] B. Ristic and B.-N. Vo. "Sensor control for multi-object state-space estimation using random finite sets". In: *Automatica* 46.11 (2010), pp. 1812–1818.

[BWT11] Y. Bar-Shalom, P. K. Willett, and X. Tian. *Tracking and Data Fusion: A Handbook of Algorithms*. Storrs, CT: YBS Publishing, 2011. ISBN: 978-0964831278.

[Cha+11] S. Challa, M. R. Morelande, D. Mušicki, and R. J. Evans. *Fundamentals of Object Tracking*. Cambridge, United Kingdom: Cambridge University Press, 2011, p. 392. ISBN: 978-0521876285.

[HC11] A. O. Hero III and D. Cochran. "Sensor Management: Past, Present, and Future". In: *IEEE Sensors Journal* 11.12 (Dec. 2011), pp. 3064–3075. ISSN: 1530-437X. DOI: 10.1109/JSEN.2011.2167964.

[LW11] W. Liu and C. Wen. "A linear multisensor PHD filter using the measurement dimension extension approach". In: *International Conference in Swarm Intelligence*. Springer. 2011, pp. 486–493.

[Pac11] M. Pace. "Stochastic models and methods for multi-object tracking". PhD thesis. Talence, France: Université de Bordeaux, École Doctorale de Mathématiques et Informatique, 2011.

[RVC11] B. Ristic, B.-N. Vo, and D. Clark. "A Note on the Reward Function for PHD Filters with Sensor Control". In: *IEEE Transactions on Aerospace and Electronic Systems* 47.2 (2011), pp. 1521–1529.

[Ris+11] B. Ristic, B.-N. Vo, D. Clark, and B.-T. Vo. "A Metric for Performance Evaluation of Multi-Target Tracking Algorithms". In: *IEEE Transactions on Signal Processing* 59.7 (July 2011), pp. 3452–3457. DOI: 10.1109/TSP.2011.2140111.

[WHE11] Y. Wang, I. Hussein, and R. S. Erwin. "Risk-Based Sensor Management for Integrated Detection and Estimation". In: *American Control Conference*. June 2011, pp. 3633–3638. DOI: 10.1109/ACC.2011.5991348.

[Del12] E. Delande. "Multi-sensor PHD Filtering with Application to Sensor Management". PhD thesis. Lille, France: Ecole Centrale de Lille, Laboratoire d'Automatique, Génie Informatique et Signal, 2012.

[ME12] R. P. S. Mahler and A. El-Fallah. "An approximate CPHD filter for superpositional sensors". In: *Signal Processing, Sensor Fusion, and Target Recognition XXI*. Vol. 8392. International Society for Optics and Photonics. 2012, 83920K.

[MDJ12] P. D. Moral, A. Doucet, and A. Jasra. "On adaptive resampling strategies for sequential Monte Carlo methods". In: *Bernoulli* 18.1 (2012), pp. 252–278.

[NCM13] S. Nannuru, M. Coates, and R. P. S. Mahler. "Computationally-tractable approximate PHD and CPHD filters for superpositional sensors". In: *IEEE Journal of Selected Topics in Signal Processing* 7.3 (2013), pp. 410–420.

[Pap+13] F. Papi, B.-T. Vo, M. Bocquel, and B.-N. Vo. "Multi-Target Track-Before-Detect using Labeled Random Finite Set". In: *Control, Automation and Information Sciences (ICCAIS), 2013 International Conference on*. IEEE. 2013, pp. 116–121.

[VV13] B.-T. Vo and B.-N. Vo. "Labeled Random Finite Sets and Multi-Object Conjugate Priors". In: *IEEE Transactions on Signal Processing* 61.13 (2013), pp. 3460–3475.

[GA14] M. S. Grewal and A. P. Andrews. *Kalman Filtering: Theory and Practice Using MATLAB*. USA: John Wiley & Sons, 2014, p. 640. ISBN: 978-1118851210.

[HV14] H. G. Hoang and B.-T. Vo. "Sensor management for multi-target tracking via multi-Bernoulli filtering". In: *Automatica* 50.4 (2014), pp. 1135–1142.

[Mah14] R. P. S. Mahler. *Advances in statistical Multisource-Multitarget Information Fusion*. Artech House, 2014. ISBN: 978-1-60807-798-4.

[Pap+14] F. Papi, B.-N. Vo, B.-T. Vo, C. Fantacci, and M. Beard. "Generalized Labeled Multi-Bernoulli Approximation of Multi-Object Densities". In: *ArXiv e-prints* (Dec. 2014).

[PPH14] Y. Punchihewa, F. Papi, and R. Hoseinnezhad. "Multiple Target Tracking in video data using Labeled Random Finite Set". In: *Control, Automation and Information Sciences (ICCAIS), 2014 International Conference on*. IEEE. 2014, pp. 13–18.

[Reu+14] S. Reuter, B.-T. Vo, B.-N. Vo, and K. Dietmayer. "The Labeled Multi-Bernoulli Filter". In: *IEEE Trans. Signal Processing* 62.12 (2014), pp. 3246–3260.

[Ros14] S. M. Ross. *Introduction to Probability Models*. Academic press, 2014.

[vH14] T. van Erven and P. Harremoës. "Rényi divergence and Kullback-Leibler divergence". In: *IEEE Transactions on Information Theory* 60.7 (2014), pp. 3797–3820.

[VVP14] B.-N. Vo, B.-T. Vo, and D. Phung. "Labeled Random Finite Sets and the Bayes Multi-Target Tracking Filter". In: *IEEE Transactions on Signal Processing* 62.24 (2014), pp. 6554–6567.

[BVV15] M. Beard, B.-T. Vo, and B.-N. Vo. "Bayesian Multi-Target Tracking With Merged Measurements Using Labelled Random Finite Sets." In: *IEEE Trans. Signal Processing* 63.6 (2015), pp. 1433–1447.

[Bea+15] M. Beard, B.-T. Vo, B.-N. Vo, and S. Arulampalam. "Sensor control for multi-target tracking using Cauchy-Schwarz divergence". In: *Information Fusion (Fusion), 2015 18th International Conference on*. IEEE. 2015, pp. 937–944.

[Hoa+15] H. G. Hoang, B.-N. Vo, B.-T. Vo, and R. P. S. Mahler. "The Cauchy-Schwarz divergence for Poisson point processes". In: *IEEE Transactions on Information Theory* 61.8 (2015), pp. 4475–4485.

[HVV15] H. G. Hoang, B.-T. Vo, and B.-N. Vo. "A fast implementation of the generalized labeled multi-Bernoulli filter with joint prediction and update". In: *Information Fusion (Fusion), 2015 18th International Conference on*. IEEE. 2015, pp. 999–1006.

[Mar15] S. R. Martin. "Risk-based Sensor Resource Management for Field of View Constrained Sensors". In: *18th International Conference on Information Fusion (Fusion), 2015*. July 2015, pp. 2041–2048.

[PK15] F. Papi and D. Y. Kim. "A Particle Multi-Target Tracker for Superpositional Measurements using Labeled Random Finite Sets". In: *IEEE Transactions on Signal Processing* 63.16 (2015), pp. 4348–4358.

[Gom+16] M. E. Gomes-Borges, D. Maltese, P. Vanheeghe, G. Sella, and E. Duflos. "Sensor Management using Expected Risk Reduction approach". In: *Information Fusion (Fusion), 2016 19th International Conference on*. IEEE. 2016, pp. 2050–2058.

[RBF16] B. Ristic, M. Beard, and C. Fantacci. "An Overview of Particle Methods for Random Finite Set Models". In: *Information Fusion* 31 (2016), pp. 110–126.

[Gom+17] M. E. Gomes-Borges, D. Maltese, P. Vanheeghe, and E. Duflos. "Risk-Based Sensor Management using Random Finite Sets". In: *Information Fusion (Fusion), 2017 20th International Conference on*. IEEE. 2017.

[Mah17] R. P. S. Mahler. "On CPHD Filters with Track Labeling". In: *Signal Processing, Sensor/Information Fusion, and Target Recognition XXVI*. Vol. 10200. International Society for Optics and Photonics. 2017, 102000E.

[SLC17] A.-A. Saucan, Y. Li, and M. Coates. “Particle flow superpositional GLMB filter”. In: *Signal Processing, Sensor/Information Fusion, and Target Recognition XXVI*. Vol. 10200. International Society for Optics and Photonics. 2017, 102000F.

[VVH17] B.-N. Vo, B.-T. Vo, and H. G. Hoang. “An Efficient Implementation of the Generalized Labeled Multi-Bernoulli Filter”. In: *IEEE Transactions on Signal Processing* 65.8 (2017), pp. 1975–1987.

[Mah18] R. P. S. Mahler. “A Fast Labeled Multi-Bernoulli Filter for Superpositional Sensors”. In: *Signal Processing, Sensor/Information Fusion, and Target Recognition XXVII*. Vol. 10646. International Society for Optics and Photonics. 2018, 106460E.

[Sch+18] I. Schlangen, E. D. Delande, J. Houssineau, and D. E. Clark. “A second-order PHD filter with mean and variance in target number”. In: *Transactions on Signal Processing* 66 (2018), pp. 48–63.

Détermination et implémentation temps-réel de stratégies de gestion de capteurs pour le pistage multi-cibles

Les systèmes de surveillance modernes doivent coordonner leurs stratégies d'observation pour améliorer l'information obtenue lors de leurs futures mesures afin d'estimer avec précision les états des objets d'intérêt (emplacement, vitesse, apparence, etc.). Par conséquent, la gestion adaptative des capteurs consiste à déterminer les stratégies de mesure des capteurs exploitant les informations *a priori* afin de déterminer les actions de détection actuelles. L'une des applications la plus connue de la gestion des capteurs est le suivi multi-objets, qui fait référence au problème de l'estimation conjointe du nombre d'objets et de leurs états ou trajectoires à partir de mesures bruitées. Cette thèse porte sur les stratégies de gestion des capteurs en temps réel afin de résoudre le problème du suivi multi-objets dans le cadre de l'ensemble aléatoire fini labélisé ou « Labeled Random Finite Set (LRFS) ». La première contribution est la formulation théorique rigoureuse du filtre mono-capteur LPHD « Labeled Probability Hypothesis Density » avec son implémentation Gaussienne. La seconde contribution est l'extension du filtre LPHD pour le cas multi-capteurs. La troisième contribution est le développement de la méthode de gestion de capteurs basée sur la minimisation du risque bayésien et formulé dans les cadres LRFS et POMDP « Partially Observable Markov Decision Process ».

Mots-clés : Gestion de Capteur, Pistage Multi-Cibles, PHD labelisée, PHD labelisée multi-capteur

---

Real-Time Sensor Management Strategies for Multi-Object Tracking

Modern surveillance systems must coordinate their observation strategies to enhance the information obtained by their future measurements in order to accurately estimate the states of objects of interest (location, velocity, appearance, etc). Therefore, adaptive sensor management consists of determining sensor measurement strategies that exploit *a priori* information in order to determine current sensing actions. One of the most challenging applications of sensor management is the multi-object tracking, which refers to the problem of jointly estimating the number of objects and their states or trajectories from noisy sensor measurements. This thesis focuses on real-time sensor management strategies formulated in the POMDP framework to address the multi-object tracking problem within the LRFS approach. The first key contribution is the rigorous theoretical formulation of the mono-sensor LPHD filter with its Gaussian-mixture implementation. The second contribution is the extension of the mono-sensor LPHD filter for superpositional sensors, resulting in the theoretical formulation of the multi-sensor LPHD filter. The third contribution is the development of the Expected Risk Reduction (ERR) sensor management method based on the minimization of the Bayes risk and formulated in the POMDP and LRFS framework. Additionally, analyses and simulations of the existing sensor management approaches for multi-object tracking, such as Task-based, Information-theoretic, and Risk-based sensor management, are provided.

Keywords: Sensor Management, Multi-Object Tracking, Labeled PHD filter, multi-sensor Labeled PHD filter