NTNU
Kunnskap for en bedre verden

DEPARTMENT OF ENGINEERING CYBERNETICS

TTK4900 - MASTER THESIS

Poisson multi-Bernoulli mixture filter for multiple extended object tracking of maritime vessels using Lidar and Gaussian processes

Author:
Michael Ernesto López Lehmann

Supervisor:
Edmund Førland Brekke

August, 2020

Abstract

An essential task of collision avoidance systems for autonomous ships is to, based on high-resolution sensor measurements and other navigational information, detect other objects in the surroundings, and to track their movement (i.e., their position, velocity and heading), as well as to estimate their extent (i.e., size and shape). This situational awareness problem is addressed by multiple extended object tracking (MEOT).

This master thesis presents a self-contained derivation of the Poisson multi-Bernoulli mixture (PMBM) filter for MEOT, which is one of the state-of-the-art methods for MEOT. Furthermore, a state-space model that uses Gaussian processes to model a wide variety of object extends and the lidar measurements that these objects generate, is integrated in the PMBM filter.

The PMBM filter and the state-space model are tested together under different simulations, where the objects resemble the hull of ships in the horizontal plane.

Samandrag

Ein hovuddel av system for kollisjonsforhindring for autonome skip er å oppdaga andre objekt i omgivnadene, og å følgja rørsla deira (dvs. posisjon, fart og kurs), i tillegg til å estimera omfanga deira (dvs. storleik og form), basert på sensormålingar med høg oppløysing og annan navigasjonsinformasjon. Utvidet følging av fleire objekt (”Multiple extended object tracking (MEOT)” på engelsk) løyser dette situasjonsmedvitsproblemet.

Denne masteroppgåva presenterer ei sjølvstendig avleiing av Poisson multi-Bernoulli-blanding filteret (”Poisson multi-Bernoulli mixture (PMBM) filter” på engelsk) for MEOT, som er ein av dei mest moderne metodane for MEOT. Vidare blir det presentert ein tilstandsrommodell som bruker Gaussiske prosessar for å modellera eit breitt utval av omfang, og lidarmålingane som desse objekta genererer.

PMBM-filteret og tilstandsrommodellen er testa saman under ulike simuleringer, der objekta liknar på skrog til skip i det horisontale planet.

Preface

This master thesis constitutes the culmination of my masters degree in Cybernetics and Robotics at the Norwegian University of Science and Technology (NTNU).

The main topic of this work is multiple extended object tracking (MEOT) for autonomous ships, which is a current and relevant topic in both industry and research, as well as a topic of personal interest. This thesis has given me the opportunity to delve into the theory of one of the most sophisticated and mathematically elegant filters for MEOT: The Poisson multi-Bernoulli mixture filter, and to integrate this filter with Gaussian Processes (GPes), which are used to derive a versatile stochastic model for the object's extent.

The work presented in this report has been completed independently, and it is the continuation of my specialization project, whose topic was single object tracking using GPes and lidar measurements. Therefore, the sections of this report about the relevant literature, GPes and the used state-space model bear great resemblance to their respective counterparts in my specialization project.

I would like to thank my supervisor, Edmund Førland Brekke, for all the advice he has given me during this project.

Furthermore, I would like to thank Yuxuan Xia for publishing his implementation of the PMBM filter for MEOT on Github (see [41]). This code was modified for the particular models presented in this thesis. Without access to this code or similar, it would have been impossible for me to both delve into the theoretical foundations of this filter and to produce results due to time constraints.

Michael Ernesto López Lehmann

August 9th, 2020

Table of Contents

Chapter 1

Introduction

1.1 Background and motivation

Autonomous surface vehicles (ASVs) that work with support from a remote control center (RCC) or in a controlled environment, are already a part of the present and their use in the industry is expected to increase in the future. In Norway alone, there are several ASV projects, such as the container ship Yara Birkeland from Kongsberg and Yara, the container ship ReVolt from DNV-GL [4] and several small ASVs produced by Maritime Robotics.

There are diverse reasons for this development. The evident economical motivation is to reduce personnel costs and the logistical costs related to having a crew on board. Furthermore, the total or partial elimination of a crew will reduce accidents at sea. In the case of container ships, the freed space by the absent crew could be used to transport more goods. Moreover, autonomous container ships are expected to take a considerably share of the freight-transport performed exclusively on land. This will reduce CO_2-emissions and alleviate traffic.

Although ASVs with partial autonomy are a reality, fully ASV are still a research topic. There are many technological and juridical challenges that need to be addressed before fully ASV can operate as usual vessels do (see for example the articles [10] and [33]). One of these technological challenges is to develop a collision avoidance system that ensures a safe operation of the vessel with the same performance of a well-trained human crew, or better.

Figure 1.1 illustrates the structure of collision avoidance systems for autonomous ships. In the ”sensor fusion module”, the information provided by charts, global navigation satellite systems (GNSSs) and inertial measurement units (IMUs) are used to determine the navigational situation of the ship, i.e., its position, velocity and heading. By fusing this information together with measurements provided by imaging sensors, such as camera, radar or lidar, the objects in the surroundings of the ship can be detected, and their movement, .i.e., their position, velocity and heading, can be tracked. The objects of interest are mostly other ships, but identifying traffic signs and landmarks such as light houses, could also improve the overall situational awareness. Automatic identification system (AIS) signals can also be used to improve the situational awareness of the ASV. However, not all maritime vessels have an AIS installed, and if they do, the broadcasted information can be very inaccurate.

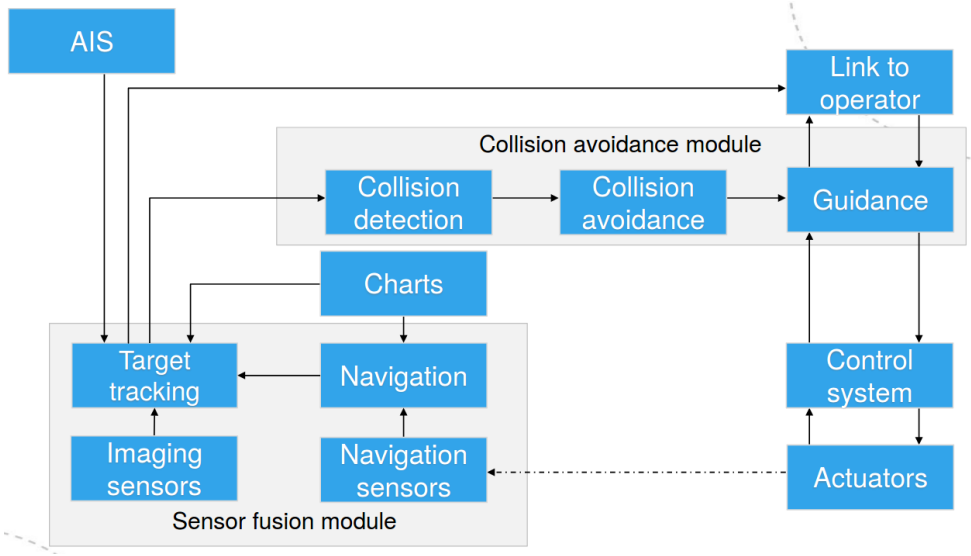


Figure 1.1: Structure of collision avoidance systems for autonomous ships. Image courtesy of [8].

The situational awareness provided by the ”sensor fusion module” is passed down to the ”collision avoidance module”, which determines whether additional maneuvers have to be undertaken to avoid a possible collision. In such case, the chosen maneuvers are transformed by the vehicle guidance controllers into control signals for the ship actuators.

The research and development of collision avoidance systems for autonomous ships are challenging endeavours. Here at NTNU, there are several related research projects on this topic: The Autoferry project [1], which develops an autonomous electric ferry for pedestrian and cyclist transport across a canal in Trondheim, the Autosea project [2], which conducts research into sensor fusion and collision avoidance methods for autonomous ships, and the Autosit project [3], which aims to improve the situational awareness and to estimate the intentions of other vessels using imaging sensors and other available information, such as AIS or charts. In particular, this master thesis is connected to the Autosea project [2], and its topic is inscribed in the submodule ”target tracking” in Figure 1.1.

Target tracking or object tracking (OT) is the problem of detecting objects and tracking their movement, i.e., estimating their kinematic properties, such as position, velocity and orientation, based on measurements. A measurement may of course be caused by a particular object, but it may also be a false alarm, which is also known as clutter. The terms single object tracking and multiple object tracking (MOT) are also used to emphasize whether one or multiple objects are considered. Furthermore, if the resolution of the sensors on-board is high enough, as is the case of modern imaging sensors, several measurements of any tracked object could be available at any given time. In this case, it is possible to also estimate the geometry of the object, also know as the object’s extent. The resulting problem is then referred to as extended object tracking (EOT) or, if multiple objects are considered, as multiple extended object tracking (MEOT).

Knowledge about the object’s extent could drastically improve the situational awareness of the collision avoidance system. For example, a large object could hide smaller undetected objects behind it. Moreover, the size of an object can tell us something about their possible

movements since larger objects are not as agile as smaller ones.

Traditionally, the extent of objects has been approximated by ellipses or by ellipsoids depending whether the problem is stated in two or three spatial dimensions. This approximation is known as the random matrix approach. Although the random matrix approach may work for airplanes and cars, it may perform poorly for maritime vessels. The main reason for this poor performance is that large parts of the hull and superstructure of a maritime vessel can have curvatures that correspond to ellipses or ellipsoids with quite different centers and semiaxes. Figure 1.2 illustrates this scenario. Here, the hull of the ship M/S Nidarholm is estimated in the horizontal plane using lidar measurements and the random matrix approach. At the initial time steps, the elliptical estimates are relatively accurate because the lidar sensor mostly illuminates a side of the hull, which has a representative curvature for the rest of the hull. However, as the estimation process progresses, the flat stern of the hull is the part that is mainly illuminated by the sensor. This causes the hull estimate to shrink to almost a circle with the same diameter as the stern width, which is an overall bad estimate of the hull.

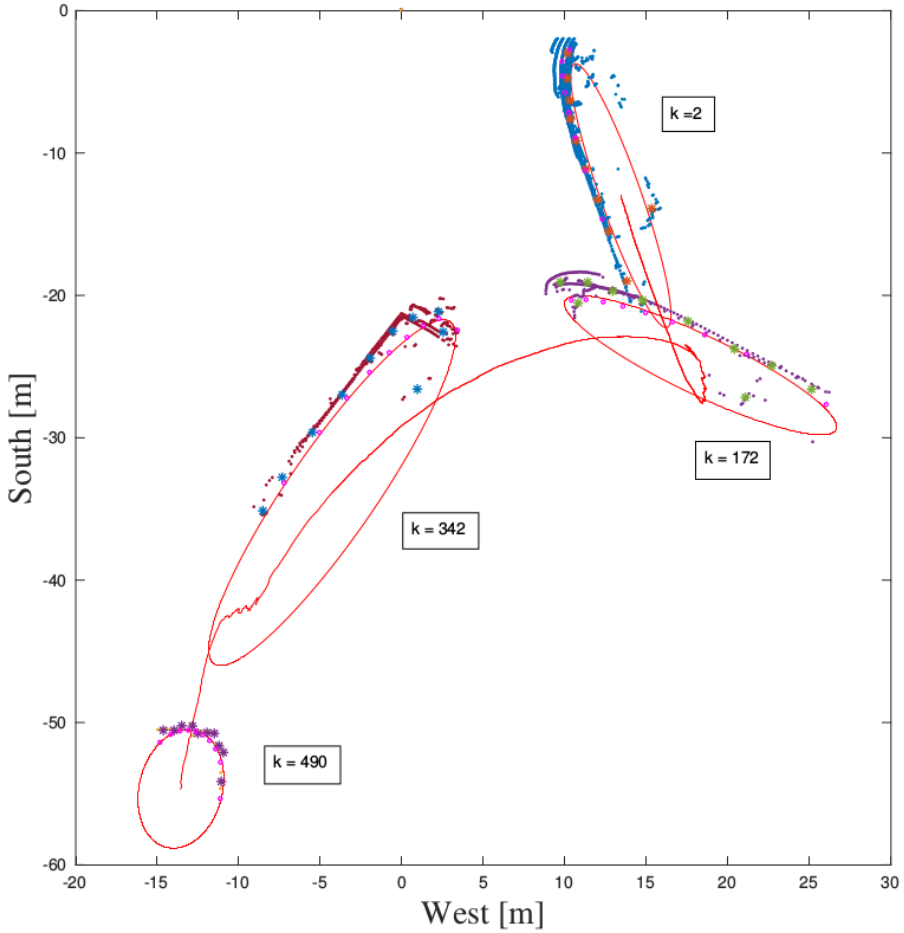
A proposed solution to the extent estimation of maritime vessels, is to model the contour or boundary of the extent as an unknown function, whose function values are distributed according to a multivariate normal distribution. This is known as a Gaussian Process (GP), which is a very versatile tool for regression and estimation. In contrast to many other approximation methods, such as the random matrix approach, GPes provide a non-parametric model in the sense that no assumptions are made on the underlying structure of the unknown function. Furthermore, this model is stochastic, and can therefore be smoothly merged with probabilistic filters, which are well represented in EOT. This enables integration of prior and uncertainty information.

EOT problems can be very diverse: As mentioned above, they can consider only a single object or multiple objects, in two or three spatial dimensions. Moreover, the existence of the objects can be known in advance, or the detection of existing objects based on measurements is a part of the EOT problem. The measurements can come from a variety of sensor types, and can present clutter. In addition, the measurements can have high levels of clutter or noise, and they can correspond to erratically moving objects, which makes the estimation problem even more challenging.

In this master thesis, we consider multiple objects whose extents resemble ship hulls in the horizontal plane. These extents are modeled using GPes. The movement and extent of the objects is estimated using a state-of-the-art MEOT filter: the Poisson multi-Bernoulli mixture (PMBM) filter. This filter is capable of detecting multiple objects, track their movement and deal with clutter.

The theory behind the PMBM filter for MEOT is mathematically sophisticated, and requires the concept of random finite sets (RFS), which are finite sets whose cardinality and elements are randomly distributed. Because of all the theoretical prerequisites, papers that present this filter do not give a detailed presentation or they refer to other sources. Therefore, in this thesis, we present a self-contained derivation of the PMBM filter that is accessible from an undergraduate level in mathematics and statistics.

(a) M/S Nidarholm.



(b) Lidar measurements of M/S Nidarholm at different time points k (small dots of different colors) and the corresponding elliptical approximations of its hulls (red ellipses), as well as estimated trajectory (red line).

Figure 1.2: Estimation of the hull of M/S Nidarholm using lidar measurements and assuming the extent to be an ellipsis. Images courtesy of [31].

1.2 Objectives

The objectives of this master thesis are the following:

1. Study the literature about the topics of Gaussian processes (GPes) and extended object tracking (EOT) with specially emphasis in EOT methods that use GPes and in the Poisson multi-Bernoulli mixture (PMBM) filter.

2. Present a self-contained derivation of the PMBM filter for MEOT that is accessible from an undergraduate level in mathematics and statistics.

3. Present an object model that uses GPes to describe two-dimensional extents, and that models lidar measurements.

4. Integrate the PMBM filter for MEOT with the object model that combines GPes and lidar measurements, and test the resulting MEOT method under different simulations, where the objects represent small ships moving at sea.

1.3 Report outline

The literature review about Gaussian processes (GPes) and their use in extended object tracking (EOT), as well as the Poisson multi-Bernoulli mixture filter and related topics, is presented in Chapter 2.

Chapter 3 treats Finite Set Statistics (FISST), and introduces the necessary concepts and results to derive the Poisson multi-Bernoulli mixture (PMBM) filter for EOT, which is done in Chapter 4. The presentation of FISST and the PMBM filter is self-contained in the sense that it is accessible from an undergraduate level in mathematics and statistics, and that all statements are mathematically proven. Moreover, Chapter 4 shows the computational intractability of the PMBM filter per se in Section 4.5, and presents solutions that make the PMBM filter computationally feasible.

Chapter 5 introduces the fundamentals of GPes and their regression technique. GPes are used in Chapter 6 to give a versatile extent model in two spatial dimensions, as well as a model for lidar measurements. Moreover, the gamma Gaussian inverse Wishart (GGIW) model is briefly introduced because it is the model implemented in the PMBM code by Yuxuan Xia (see [41]). The PMBM implementation in [41] is modified to work with the GP and lidar model developed in Chapter 6. The PMBM filter for MEOT is tested under several simulations for both models in Chapter 6, and the obtained results are shown and discussed in Chapter 7.

Finally, in light of the discussed theory and results, the conclusions of this master thesis, as well as ideas for further work, are presented in Chapter 8.

In addition, some results about multivariate normal distributions that may not be part of an undergraduate course in probability and statistics is present in Appendix A. This appendix is followed by remarks on the notation used, and lists of abbreviations and the most important symbols, as well as the bibliography.

Chapter 2

Literature review

Gaussian Processes

A Gaussian Processes (GP) is a stochastic model for an unknown function that assumes that any finite subset of the unknown function values follows a multivariate normal distribution according to a common mean and covariance functions. In particular, GPes can be seen as generalizations of multivariate normal distributions.

The theoretical background for GPes and their regression technique were developed by Wiener [38] and Kolmogorov [22] in the 40's. However, the alleged first major application of GP regression can be traced back to the work of Krige [23] in 1951, who, motived by the insatisfactory results of current methods, developed an interpolating technique of geostatistical data for mining surveys. Therefore, GP regression is also known as "Wiener-Kolmogorov prediction" or "Kriging".

GPes are appeling for modeling since they are non-parametric function models, in the sense that they do not make assumptions on the underlying structure of the functions they model. In addition, GPes are stochastical models, and can therefore be smoothly integrated with other stochastical models and probabilistic filters. The traditional drawback of GPes is the high computational cost of GP regression, which usually involves the inversion of a large matrix. However, due to the continuous growth in processing power in the last decades, this drawback is no longer a major restriction, an GPes can even be used in real-time systems. As a consequence, the applications of GPes have moved from spatial statistics to many other fields, where optimization and prediction is crucial. One of these fields is Machine Learning, where the objective is to find the unknown relation between the input and output of a system. Therefore, GP provides an ideal framework to solve this problem. In this regard, the book "Gaussian Processes for Machine Learning" [30] by Rasmussen and Williams is the go-to reference in the topic, and this book served as the foundation of Chapter 5 of this report.

Object tracking

The historical origins of object tracking (OT) can be traced back to the Second World War when Allied radar stations began to be used as warning systems against enemy aircraft attacks. At these stations, radar signals where used to estimate the position and direction of the enemy objects. Since radar technology was at its infancy at that time, each object extent was approximated by a point.

The first contribution to the literature about EOT that is stated in a way that resembles the modern EOT formalism, can be traced back to 1988 in the work of Drummond et al. in [11]. Many other contributions to the field have been added since then. In the tutorial paper [19], Granström et al. formulate the EOT problem in a multiple target setting, and give an overview of the current state-of-the-art research in the field up to the article's publication in 2017. Here, the two most used EOT approaches are discussed: The random matrix approach and the extended Kalman filter (EKF) approach for star-convex shapes. Both approaches are Bayesian, and divide the object's state vector into kinematic states that describe the movement and orientation of the object, such as position, velocity and heading, and into extent states, which, as the name suggests, describe the extent of the object.

In the random matrix approach, the extent is assumed to be an ellipsis or an ellipsoid depending on whether the spatial aspect of the problem is two or three dimensional. In this model, the chosen extent states represent the symmetric and positive definite matrix that models the implicit cartesian equations of the elliptical or ellipsoidal extent.

The random matrix approach was first proposed by Koch in [21], where a Bayesian approach was used to estimate the elliptical extent. In this work, there was a restrictive coupling between the kinematic and extent states. In addition, sensor noise was not part of the model. These issues were later addressed by Feldmann et al. in [12]. In [16], Granström and Orguner improved the random matrix approximation by modeling the rotations of the elliptical extent. Furthermore, in [34], Schuster and Reuter used their general probabilistic data association (GPDA) filter (see [35]) to handle clutter measurements, and they performed experiments with both radar and lidar measurements. In [15], Granström et al. used the probability hypothesis density (PHD) filter to develop an EOT method that estimates the contour of multiple rectangular and elliptical objects using lidar measurements. Inspired by this, Ruud combined the sensor model from [15] with the GPDA approach from [35] in his MSc thesis [31] and related conference paper [32]. The developed method was tested on real lidar data measurements of the ship M/S Nidarholm, whose hull shape can be fairly approximated by an ellipse. Despite this apparent reasonable approximation, as mentioned in Section 1.1, the random matrix approach gave elliptical estimates that varied broadly in both shape and size, and that in many cases did not guard a resemblance with the actual extent. This serves as a motivation to explore the use of GPes for modeling the extent of maritime vessels.

On the other side, in the EKF approaches for star-convex shapes, the methods assume the extent to be star-convex, which is consider a mild assumption, if not a technicality, since most vehicles have star-convex shapes. The star-convex assumption ensures that the extent is univocally determined by its boundary. Furthermore, the polar parametrization of the boundary with respect to a point inside the extent and a reference frame attached to the body is well-defined

due to this assumption. Hence, the extent is univocally determined by a radius function that parameterizes the extent's boundary.

Under this framework, there are two model classes for representing the star-convex extent: The random hypersurface models and the GP models. In the random hypersurface models, the extent states correspond to the Fourier coefficients of the radius function that parameterizes the extent's boundary. The more Fourier coefficients are added to the object's state, the more detailed the estimate of the extent can be. In [7], Baum and Hanebeck present an EOT method that uses random hypersurfaces for radar measurements. On the other hand, in the GP models, the extent states correspond to a discretization of the radius function, and this unknown radius function is assumed to be distributed as a GP. Again, the finer the discretization of the radius function is, the more detailed the estimated extent can be. In [37], Wahlström and Özkan used this approach to develop an EOT method for lidar measurements. The resulting measurement model is highly non-linear, and the accurate association of measurements to points in the extent's boundary depends strongly on good position and heading estimates. In [27], Michaelis et al. propose an arguably more robust method to associate measurements to the extent's boundary using random hypersurfaces.

Finite Set Statistics and its applications to object tracking

Finite Set Statistics (FISST) is a theory that studies the stochastic properties of random finite sets (RFSs), which are finite sets whose cardinality and elements are randomly distributed. The book [26] by Mahler provides a comprehensive discussion of the topic. FISST provides a convenient model for multiple object tracking (MOT). For example, by modeling objects as RFSs, one can elegantly address the fact that there is uncertainty in the number of objects, i.e., there is uncertainty in their existence.

There are two main classes of MOT methods based on RFSs: Those that use labeled RFSs, and those that use unlabeled RFSs. In methods based on labeled RFSs, the tracked objects remain unchanged from one time-step to the next. This gives track continuity. However, since there is no order for the elements of a set, labels are introduced in order to identify each individual tracked object. A relevant example of an MEOT method based on labeled RFSs is the generalized labeled multi-Bernoulli (GLMB) presented by Hirscher et al. in [20], which uses the GP extent model and lidar measurement model introduced in [37]. On the other side, MOT methods based on unlabeled RFS do not specifically model track continuity, and the correspondence between the tracked objects at one time point and the next is not immediate to determine. However, this is a minor issue because there is always track continuity in practice due to the continuity of the underlying object model. Moreover, unlabeled RFS can be stated in a more elegant and effective frame-work than labeled RFS since they do not require the extra bookkeeping that comes with the use of labels. The Poisson multi-Bernoulli mixture (PMBM) derived by Williams in [39] is an example of an MOT method that uses unlabeled RFSs. In [14], Granström et al. expands this filter to MEOT and uses it together with a random matrix extent model and a radar measurement model.

In this master thesis, we will explore the combination of the PMBM filter for MEOT from [14] with the GP extent and lidar measurement model from [37].

Chapter 3

Finite Set Statistics

A random finite set (RFS) is the mathematical abstraction of a finite set whose cardinality and elements are randomly distributed. Finite Set Statistics (FISST) is the theory that studies the statistical properties of these random sets, and provides mathematical tools for statistical inference on them.

In MOT, the number of objects, as well as their particular characteristics, are unknown and can vary with time. Therefore, RFSs provide a convenient model for the groups of undetected and detected objects. Moreover, by utilizing RFS statistical inference, the number of objects and the values of their parameters can be estimated.

In this chapter, we introduced the basic concepts of RFS and the necessary results for deriving the Poisson multi-Bernoulli mixture (see chapter 4). The presented results are proven using results from standard -,i.e., undergraduate level- calculus and probability theory. Furthermore, it is worth noticing that most of these results resemble analogous results from standard probability theory. This may improve the intuition, and compensate for the added abstractional difficulty of studying the stochastics of sets.

For a comprehensive exposition of FISST, we refer to [26].

3.1 Random Finite Sets

Definition 3.1.1 (Random Finite Set)**.** *A random finite set (RFS) $\mathcal{X}$ on a base space $\mathbb{S}$ is a random variable whose values are finite subsets of $\mathbb{S}$. In other words, $\mathcal{X}$ is a function that maps an outcome space Ω into the set of all finite subsets of $\mathbb{S}$, $\mathcal{P}_F(\mathbb{S})$, i.e., $\mathcal{X} : \Omega \to \mathcal{P}_F(\mathbb{S})$.*

For most applications, including the ones of this thesis, the base space $\mathbb{S}$ of a RFS is chosen to be a vector space $\mathbb{R}^d$, and this will be assumed from now onwards. Therefore, any realization X of a RFS $\mathcal{X}$ is a finite subset of $\mathbb{R}^d$, i.e., $X = \emptyset$ or $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_{|X|}\}$ for some different vectors $\mathbf{x}_1, \ldots, \mathbf{x}_{|X|} \in \mathbb{R}^d$. By comparison, a random variable in standard probability and statistics is real vector valued, i.e., it always returns one vector.

3.2 Set density functions

In standard probability theory, the concepts of probability density function and probability mass functions are a convenient tool for calculating probabilities and for studying the stochastic properties of their corresponding distributions.

The RFS version of these probability functions is the set density function, also known as the multi-object density or Janossy density, which describes the statistical properties of both the cardinality and the elements of a RFS.

Definition 3.2.1 (Set density function)**.** *Let $\mathcal{X}$ be a RFS with base space $\mathbb{R}^d$. A non-negative set function $f_{\mathcal{X}} : \mathcal{P}_F(\mathbb{R}^d) \to [0, \infty)$ is a set density function for $\mathcal{X}$ if the following conditions are verified:*

C.1 *For every $n \in \mathbb{N}$, the vector function $f_n : \mathbb{R}^d \times \overset{n}{\ldots} \times \mathbb{R}^d \to [0, \infty)$ defined by*

$$f_n(\mathbf{x}_1, \ldots, \mathbf{x}_n) = \begin{cases} f_{\mathcal{X}}(\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}) & \text{, if } \mathbf{x}_i \neq \mathbf{x}_j \text{ for } i \neq j \\ 0 & \text{, otherwise} \end{cases}, \tag{3.1}$$

is either identically zero if $Pr(|\mathcal{X}| = n) = 0$, or if it is proportional to the joint probability density function $g_n(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ that is associated to the cardinality-conditioned distribution

$$\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} | |\mathcal{X}| = n \tag{3.2}$$

if $Pr(|\mathcal{X}| = n) > 0$.

C.2 *The probability that the cardinality of the RFS $\mathcal{X}$ is equal to $n \in \mathbb{N}$ is given by the integral*

$$Pr(|\mathcal{X}| = n) = \frac{1}{n!} \int_{\mathbb{R}^d} f_n(\mathbf{x}_1, \ldots, \mathbf{x}_n)\, d\mathbf{x}_1 \ldots d\mathbf{x}_n\,, \tag{3.3}$$

and the probability that the cardinality of $\mathcal{X}$ is 0, i.e., that the $\mathcal{X}$ is the empty set, is equal to $f(\emptyset)$. In symbols, $Pr(|\mathcal{X}| = 0) = Pr(|\mathcal{X}| = \emptyset) = f(\emptyset)$.

Before dwelling into the statistical meaning of conditions **C.1** and **C.2**, note that a set density function takes values on finite subsets of $\mathbb{R}^d$, and not on terns of vectors as standard probability functions do. In particular, a set density function is necessarily invariant under permutations of the set elements, i.e.,

$$f_{\mathcal{X}}(\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}) = f_{\mathcal{X}}(\{\mathbf{x}_{\sigma(1)}, \ldots, \mathbf{x}_{\sigma(n)}\}) \tag{3.4}$$

for all permutations $\sigma \in S_n$ [1]. This is because the sets $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ and $\{\mathbf{x}_{\sigma(1)}, \ldots, \mathbf{x}_{\sigma(n)}\}$ are equal. In contrast, a standard density functions does not necessarily take the same value on the terns of vectors $(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ and $(\mathbf{x}_{\sigma(1)}, \ldots, \mathbf{x}_{\sigma(n)})$.

Furthermore, it is important to observe that the set $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ has cardinality less or equal to n, and it is only equal to n if all the vectors are different. Therefore, the case definition

[1] S_n is the set of all permutations on $\{1, \ldots, n\}$, i.e., the set of all bijections $\sigma : \{1, \ldots, n\} \to \{1, \ldots, n\}$.

in (3.1) ensures that the function f_n is well-defined. However, this case definition is just a formality, since the set of all vector terns with at least two equal components:

$$\{(\mathbf{x}_1, \ldots, \mathbf{x}_n) \in \mathbb{R}^d \times \overset{n}{\ldots} \times \mathbb{R}^d : \mathbf{x}_i = \mathbf{x}_j \text{ for some } i, j \in \{1, \ldots, n\} \text{ with } i \neq j\}, \tag{3.5}$$

has zero probability, and is therefore neglegible from a probabilistic point of view. The reason for the probability being zero is that this set is one dimension less than the dimension of $\mathbb{R}^d \times \overset{n}{\ldots} \times \mathbb{R}^d$, and that the distribution of $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ given $|\mathcal{X}| = n$ is continuous since it is given by a probability density function according to condition **C.1**.

Condition **C.1** provides the marginal distribution of the elements of $\mathcal{X}$ for a given cardinality, while condition **C.2** gives the distribution of the cardinality of the RFS $\mathcal{X}$, $|\mathcal{X}|$, which is by definition discrete because it takes values in $\mathbb{N} \cup \{0\}$.

The integral on the right-hand side of (3.3) in condition **C.2** accumulates the contribution of each possible permutation of set elements in the argument of f_n. As sets are invariant under permutations of their elements, all these contributions are equal. Therefore, since there are $n!$ possible permutations of n elements, this integral is multiplied by the factor $\frac{1}{n!}$ in order to compensate for the $n!$ identical contributions.

The following result proves that the proportionality constant mentioned in condition **C.1** is equal to $n!\Pr(|\mathcal{X}| = n)$, and gives an explicit expression for the set density function.

Proposition 3.2.1. *(Explicit expression for set density function) The proportionally constant in condition **C.1** of Definition 3.1.1 is equal to $n!Pr(|\mathcal{X}| = n)$ for $n \in \mathbb{N}$ with $Pr(|\mathcal{X}| = n) \neq 0$.*

In particular, the set density function $f_{\mathcal{X}}$ can be rewritten as

$$f_{\mathcal{X}}(X) = \begin{cases} Pr(\mathcal{X} = \emptyset) & , \text{if } X = \emptyset \\ n!Pr(|\mathcal{X}| = n)\, f_{\mathcal{X},n}(\mathbf{x}_1, \ldots, \mathbf{x}_n) & , \text{if } |X| = n \text{ and } X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}, \end{cases} \tag{3.6}$$

where $f_{\mathcal{X},n}$ is identically zero if $Pr(|\mathcal{X}| = n) = 0$, or it is the joint probability density function of the cardinality-conditioned distribution $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} || \mathcal{X}| = n$ if $Pr(|\mathcal{X}| = n) > 0$.

Proof. Let $n \in \mathbb{N}$ with $\Pr(|\mathcal{X}| = n) \neq 0$, and let c_n denote the corresponding proportionality constant. Hence, by condition **C.1**, $f_n = c_n g_n$, where f_n is defined as $f_n(\mathbf{x}_1, \ldots, \mathbf{x}_n) = f_{\mathcal{X}}(\{\mathbf{x}_1, \ldots, \mathbf{x}_n\})$ and g_n is the joint probability density function of the cardinality-conditioned distribution $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} || \mathcal{X}| = n$.

Since probability density functions integrate to 1, it follows from (3.3) in condition **C.2** that

$$\Pr(|\mathcal{X}| = n) = \frac{1}{n!} \int_{\mathbb{R}^d} f_n(\mathbf{x}_1, \ldots, \mathbf{x}_n)\, d\mathbf{x}_1 \ldots d\mathbf{x}_n \tag{3.7a}$$

$$= \frac{1}{n!} \int_{\mathbb{R}^d} c_n g_n(\mathbf{x}_1, \ldots, \mathbf{x}_n)\, d\mathbf{x}_1 \ldots d\mathbf{x}_n = \frac{1}{n!} c_n \tag{3.7b}$$

$$\Rightarrow c_n = n!\Pr(|\mathcal{X}| = n), \tag{3.7c}$$

as we sought to prove.

In particular, for any finite set $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ with n elements, it follows that

$$\begin{aligned} f_{\mathcal{X}}(X) &= f_{\mathcal{X}}(\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}) && \text{(3.8a)} \\ &= f_n(\mathbf{x}_1, \ldots, \mathbf{x}_n) && \text{(3.8b)} \\ &= n!\Pr(|\mathcal{X}| = n)\, g_n(\mathbf{x}_1, \ldots, \mathbf{x}_n) && \text{(3.8c)} \\ &= n!\Pr(|\mathcal{X}| = n)\, f_{\mathcal{X},n}(\mathbf{x}_1, \ldots, \mathbf{x}_n) && \text{(3.8d)} \end{aligned}$$

by the definitions of f_n, g_n and $f_{\mathcal{X},n}$.

Moreover, this equality is trivially valid for $n \in \mathbb{N} \cup \{0\}$ with $\Pr(|\mathcal{X}| = n) = 0$ because $f_{\mathcal{X}}(X)$ is identically zero for sets with n elements if n is such that $\Pr(|\mathcal{X}| = n) = 0$, as stated in condition **C.1**.

Hence, the set density function $f_{\mathcal{X}}$ can be rewritten as

$$f_{\mathcal{X}}(X) = \begin{cases} \Pr(\mathcal{X} = \emptyset) & \text{, if } X = \emptyset \\ n!\Pr(|\mathcal{X}| = n)\, f_{\mathcal{X},n}(\mathbf{x}_1, \ldots, \mathbf{x}_n) & \text{, if } |X| = n \text{ and } X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}. \end{cases} \tag{3.9}$$

□

3.3 Independence of random finite sets

Definition 3.3.1 (RFS independence)**.** *Two RFSs $\mathcal{X}$ and $\mathcal{Y}$ are said to be independent if*

$$Pr(\mathcal{X} \subset X \text{ and } \mathcal{Y} \subset Y) = Pr(\mathcal{X} \subset X)\, Pr(\mathcal{Y} \subset Y)\ , \tag{3.10}$$

or more formally,

$$\begin{aligned} &Pr(\{\omega \in \Omega : \mathcal{X}(\omega) \subset X \text{ and } \mathcal{Y}(\omega) \subset Y\}) \\ &= Pr(\{\omega \in \Omega : \mathcal{X}(\omega) \subset X\})\, Pr(\{\omega \in \Omega : \mathcal{Y}(\omega) \subset Y\}) \end{aligned} \tag{3.11}$$

for all subsets X and Y of the base spaces of $\mathcal{X}$ and $\mathcal{Y}$, respectively.

Moreover, a finite number of RFSs are said to be independent if any two of them are independent of each other.

The definition of independence for RFSs is very similar to the one for random variables in standard probability theory. Moreover, the well-known convolution expression for the probability density or mass function of a random variable that is the sum of independent random variables, has its RFS version using set density functions.

Theorem 3.3.1. *(Union of independent RFS) Let $\mathcal{X}_1, \ldots, \mathcal{X}_n$ be a finite number of independent RFSs with the same base space $\mathbb{R}^d$. Then the RFS $\mathcal{X}$ defined as $\mathcal{X} = \bigcup_{i=1}^{n} \mathcal{X}_i$ is given by set density function*

$$f_{\mathcal{X}}(X) = \sum_{\biguplus_{i=1}^{n} X_i = X} \prod_{i=1}^{n} f_{\mathcal{X}_i}(X_i)\,, \tag{3.12}$$

where the notation $\biguplus_{i=1}^{n} X_i = X$ denotes that $X_1, \ldots, X_n$ is a partition of X, i.e., $\biguplus_{i=1}^{n} X_i = X$ and $X_i \cap X_j = \emptyset$ for $i \neq j$. Hence, the summation in (3.12) goes through all the ways one can split the set X into the sets $X_1, \ldots, X_n$, and it adds the product of the corresponding set density function values.

Proof. We proof the expression for $f_{\mathcal{X}}(X)$ by using the identity proven in Proposition 3.2.1.

If $X = \emptyset$ and $\biguplus_{i=1}^{n} X_i = X$, then $X_i = \emptyset$ for all i is the only possible partition of X. Hence, it follows from Proposition 3.2.1, the independence of the RFSs $\mathcal{X}_i$ and the definition of $\mathcal{X}$ that

$$\begin{aligned} f_{\mathcal{X}}(X) &= f_{\mathcal{X}}(\emptyset) && \text{(3.13a)} \\ &= \text{Pr}\left(\mathcal{X} = \emptyset\right) && \text{(3.13b)} \\ &= \prod_{i=1}^{n} \text{Pr}\left(\mathcal{X}_i = \emptyset\right) && \text{(3.13c)} \\ &= \sum_{\biguplus_{i=1}^{n} X_i = X} \prod_{i=1}^{n} f_{\mathcal{X}_i}(X_i). && \text{(3.13d)} \end{aligned}$$

Assume now that $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$ with $|X| = m$. Since $\mathcal{X} = \bigcup_{i=1}^{n} \mathcal{X}_i$, by using combinatorics and the independence of the RFSs $\mathcal{X}_i$, the probability density function for the conditional distribution

$$\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} | |\mathcal{X}| = m \tag{3.14}$$

can be expressed in terms of the conditional probability functions $f_{\mathcal{X},m}$ as

$$f_{\mathcal{X},m}(\mathbf{x}_1, \ldots, \mathbf{x}_m) = \sum_{\biguplus_{i=1}^{n} X_i = X} \frac{1}{m! \text{Pr}\left(|\mathcal{X}| = m\right)} \prod_{i=1}^{n} |X_i|! \text{Pr}\left(|\mathcal{X}_i| = |X|_i\right) f_{\mathcal{X}_i,|X_i|}(\mathbf{x}_1^i, \ldots \mathbf{x}_{|X_i|}^i), \tag{3.15}$$

where $\mathbf{x}_1^i, \ldots \mathbf{x}_{|X_i|}^i$ are the elements of X_i, and $f_{\mathcal{X}_i,0}$ is taken as identically 1. Note that the factors $m!$ and $|X_i|!$ come from all the possible permutations of the elements in X and X_i, respectively. Moreover, the marginal probability density functions $f_{\mathcal{X}_i,|X_i|}$ require to be scaled with $\text{Pr}\left(|\mathcal{X}_i| = |X|_i\right)$.

Hence, it follows from Proposition 3.2.1 that

$$\begin{aligned} f_{\mathcal{X}}(X) &= m! \text{Pr}\left(|\mathcal{X}| = m\right) f_{\mathcal{X},m}(\mathbf{x}_1, \ldots, \mathbf{x}_m) && \text{(3.16a)} \\ &= m! \text{Pr}\left(|\mathcal{X}| = m\right) \sum_{\biguplus_{i=1}^{n} X_i = X} \frac{1}{m! \text{Pr}\left(|\mathcal{X}| = m\right)} \prod_{i=1}^{n} |X_i|! \text{Pr}\left(|\mathcal{X}_i| = |X|_i\right) f_{\mathcal{X}_i,|X_i|}(\mathbf{x}_1^i, \ldots \mathbf{x}_{|X_i|}^i) && \text{(3.16b)} \\ &= \sum_{\biguplus_{i=1}^{n} X_i = X} \prod_{i=1}^{n} |X_i|! \text{Pr}\left(|\mathcal{X}_i| = |X|_i\right) f_{\mathcal{X}_i,|X_i|}(\mathbf{x}_1^i, \ldots \mathbf{x}_{|X_i|}^i) && \text{(3.16c)} \\ &= \sum_{\biguplus_{i=1}^{n} X_i = X} \prod_{i=1}^{n} f_{\mathcal{X}_i}(X_i). && \text{(3.16d)} \end{aligned}$$

□

3.4 Some important random finite set classes

The relevant classes of RFSs for the PMBM filter (see chapter 4) are all given by set density functions, and are introduced here. These presentations serve also as examples for the RFS concepts that have been developed so far.

3.4.1 Poisson Point Processes

A Poisson point process (PPP) is a RFS whose cardinality is Poisson distributed and whose set elements are independent and identically distributed according to a probability density function.

Hence, if $\mathcal{X}$ is a PPP, then its cardinality distribution is given by

$$\Pr\left(|\mathcal{X}| = n\right) = \frac{e^{-\lambda}\lambda^n}{n!} \tag{3.17}$$

while the joint probability density function $f_{\mathcal{X},n}(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ for the cardinality-conditioned set distribution $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}||\mathcal{X} = n$ is

$$f_{\mathcal{X},n}(\mathbf{x}_1, \ldots, \mathbf{x}_n) = \prod_{i=1}^{n} f(\mathbf{x}_i)\,, \tag{3.18}$$

where $\lambda > 0$ is the Poisson rate and $f : \mathbb{R}^d \to [0, \infty)$ is the probability density function for any set element, also known as the spatial distribution.

In particular, it follows from (3.6) in Proposition 3.2.1 that the set density function for the PPP $\mathcal{X}$ is

$$f_{\mathcal{X}}(X) = \begin{cases} e^{-\lambda} & \text{, if } X = \emptyset \\ n!\frac{e^{-\lambda}\lambda^n}{n!}\prod_{i=1}^{n} f(\mathbf{x}_i) & \text{, if } |X| = n \text{ and } X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}. \end{cases} \tag{3.19a}$$

$$= e^{-\lambda}\lambda^{|X|}\prod_{\mathbf{x}\in X} f(\mathbf{x})\,, \tag{3.19b}$$

where the convention $\prod_{\mathbf{x}\in\emptyset} f(\mathbf{x}) = 1$ is used.

A PPP is then completely determined by its Poisson rate and spatial distribution. Moveover, the number of parameters needed to represent a PPP, can be further reduced to one by using the intensity function, which is defined as $D(\mathbf{x}) = \lambda f(\mathbf{x})$. Indeed, one can recover λ and $f(\mathbf{x})$ from $D(\mathbf{x})$ by using the fact that $f(\mathbf{x})$ integrates up to 1. Hence, $\lambda = \int_{\mathbb{R}^d} D(\mathbf{x})\, d\mathbf{x}$ and $f(\mathbf{x}) = \frac{D(\mathbf{x})}{\lambda}$.

Another important property of the intensity function is that its integral over a set $S \subset \mathbb{R}^d$, $\int_S D(\mathbf{x})\, d\mathbf{x}$, is the expected number of elements in $\mathcal{X} \cap S$. This is because λ is the expected number of elements in the set $\mathcal{X}$, and $\int_S f(\mathbf{x})\, d\mathbf{x}$ is the probability that an element is in S.

Moreover, by using the intensity function, the PPP's set density function can be elegantly rewritten as

$$f_{\mathcal{X}}(X) = e^{-\lambda}\lambda^{|X|}\prod_{\mathbf{x}\in X} f(\mathbf{x}) \tag{3.20}$$

$$= e^{-\lambda}\prod_{\mathbf{x}\in X}\lambda f(\mathbf{x}) = e^{-<D,1>}\prod_{\mathbf{x}\in X} D(\mathbf{x})\,, \tag{3.21}$$

where $< f, g >= \int_{\mathbb{R}^d} f(\mathbf{x})g(\mathbf{x})\,d\mathbf{x}$ denotes the scalar product of the functions f and g.

In the PMBM filter, PPPs are used to model clutter measurements, object-generated measurements, undetected objects and the birth of new objects.

3.4.2 Bernoulli Processes

A Bernoulli process is a RFS whose cardinality follows a Bernoulli distribution and whose set elements are distributed according to a probability density function.

Hence, according to (3.6) in Proposition 3.2.1, the set density function of a Bernoulli process $\mathcal{X}$ has the form

$$f_{\mathcal{X}}(X) = \begin{cases} 1-r & \text{, if } X = \emptyset \\ rf(\mathbf{x}) & \text{, if } X = \{\mathbf{x}\} \\ 0 & \text{, if } |X| \geq 2\,, \end{cases} \tag{3.22}$$

where r is the probability that the cardinality of $\mathcal{X}$ is 1, and $f(\mathbf{x})$ is the probability density function of the only set element in that case.

Bernoulli processes give an elegant representation of single objects in MOT because they model both the uncertainty in the object's existence by means of the parameter r, and the uncertainty in the object's state $\mathbf{x}$ by means of the probability density function $f(\mathbf{x})$. Because of this the parameter r is also referred to as the existence probability.

3.4.3 Multi-Bernoulli Processes

A multi-Bernoulli (MB) process $\mathcal{X}$ is the union of a finite number of independent Bernoulli processes $\{\mathcal{X}_i\}_{i\in\mathbb{I}}$. Hence,

$$\mathcal{X} = \biguplus_{i\in\mathbb{I}} \mathcal{X}_i\,, \tag{3.23}$$

and it follows from (3.12) in Theorem 3.3.1 that the set density function of $\mathcal{X}$ is

$$f_{\mathcal{X}}(X) = \sum_{\uplus_{i\in\mathbb{I}} X_i = X}\prod_{i\in\mathbb{I}} f_{\mathcal{X}_i}(X_i)\,, \tag{3.24}$$

where

$$f_{\mathcal{X}_i}(X_i) = \begin{cases} 1-r^i & \text{, if } X_i = \emptyset \\ r^i f^i(\mathbf{x}_i) & \text{, if } X_i = \{\mathbf{x}_i\} \\ 0 & \text{, if } |X_i| \geq 2 \end{cases} \tag{3.25}$$

and r^i and f^i are the existence probability and the probability density function of the i-th Bernoulli component, respectively. In particular, the set of parameters

$$\{(r^i, f^i)\}_{i \in \mathbb{I}} \tag{3.26}$$

completely defines the multi-Bernoulli process $\mathcal{X}$.

Since MB processes are the union of independent Bernoulli processes, they naturally represent a group of objects in MOT. Therefore, MB processes are used in the PMBM filter to model the group of objects that have been detected and are being tracked.

3.4.4 Multi-Bernoulli Mixtures

A multi-Bernoulli mixture (MBM) is a RFS whose set density function is the weighted average of a finite number of MB set density functions. Hence, if $\mathcal{X}$ is an MBM, then its set density function is given by

$$f_{\mathcal{X}}(X) = \sum_{j \in \mathbb{J}} w^j f_{\mathcal{X}_j}(X)\,, \tag{3.27}$$

where $\mathbb{J}$ is a finite index set, w^j are weights such that $w^j \geq 0$ and $\sum_{j \in \mathbb{J}} w^j = 1$ and $f_{\mathcal{X}_j}$ is the set density function of an MB with parameters

$$\{(r^{j,i}, f^{j,i})\}_{i \in \mathbb{I}_j}. \tag{3.28}$$

In particular, an MBM is completely determined by the set of parameters

$$\{(w^j, \{(r^{j,i}, f^{j,i})\}_{i \in \mathbb{I}_j}\}_{j \in \mathbb{J}}. \tag{3.29}$$

In the PMBM filter, each weight w^j is the probability for a particular data association hypothesis (see section 4.2), and its corresponding MB $\mathcal{X}_j$ represents the group of tracked objects for that hypothesis.

3.5 The set integral

The set integral is the generalization of the standard Riemann integral to set functions, and it is defined as follows:

Definition 3.5.1 (Set integral)**.** *Let S be a subset of $\mathbb{R}^d$ and let $f : \mathcal{P}_F(S) \to \mathbb{R}$ be a set function such that for all $n \in \mathbb{N}$, the function from $S \times \overset{n}{\ldots} \times S$ to $\mathbb{R}$ defined as*

$$(\mathbf{x}_1, \ldots, \mathbf{x}_n) \to f(\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}) \tag{3.30}$$

is Riemann integrable over $S \times \overset{n}{\ldots} \times S$.

Then the set integral of f on S is defined as

$$\int_S f(X)\, \delta X = f(\emptyset) + \sum_{n=1}^{\infty} \frac{1}{n!} \int_{S \times \overset{n}{\ldots} \times S} f(\{\mathbf{x}_1, \ldots, \mathbf{x}_n\})\, d\mathbf{x}_1 \ldots d\mathbf{x}_n. \tag{3.31}$$

Moreover, if $S = \mathbb{R}^d$, we only write $\int f(X)\, \delta X$ for the integral over the whole space $\mathbb{R}^d$.

Note that similarly to (3.3), the integrals on the right-hand side of (3.31) are multiplied by the factor $\frac{1}{n!}$ in order to compensate that the function $f(\{\mathbf{x}_1, \ldots, \mathbf{x}_n\})$ is invariant under permutations of the set elements $\mathbf{x}_1, \ldots, \mathbf{x}_n$, which gives $n!$ identical contributions to its integral.

In standard probability theory, the integral of a probability density function over a set is equal to the probability that the realizations of its random variable lie in that set. The analogous result for set density functions is summarized in the following theorem:

Theorem 3.5.1. *(Probability from set integral) Let $\mathcal{X}$ be a RFS with base space $\mathbb{R}^d$ and set density function $f_{\mathcal{X}}$.*

Then the probability that the realizations of $\mathcal{X}$ are contained in a set $S \subset \mathbb{R}^d$ is equal to the set integral of its set density function on that set S, i.e.,

$$Pr\left(\mathcal{X} \subset S\right) = \int_S f_{\mathcal{X}}(X)\, \delta X. \tag{3.32}$$

Proof. By inserting the case-based expressions of $f_{\mathcal{X}}$ according to (3.6) into (3.31) and using that the integral of a probability density functions gives the probability, we obtain that

$$\int_S f_{\mathcal{X}}(X)\, \delta X = f(\emptyset) + \sum_{n=1}^{\infty} \frac{1}{n!} \int_{S \times \overset{n}{\ldots} \times S} f_{\mathcal{X}}(\{\mathbf{x}_1, \ldots, \mathbf{x}_n\})\, d\mathbf{x}_1 \ldots d\mathbf{x}_n \tag{3.33a}$$

$$= \Pr\left(|\mathcal{X}| = 0\right) + \sum_{n=1}^{\infty} \frac{1}{n!} \int_{S \times \overset{n}{\ldots} \times S} n! \Pr\left(|\mathcal{X}| = n\right) f_{\mathcal{X},n}(\mathbf{x}_1, \ldots, \mathbf{x}_n)\, d\mathbf{x}_1 \ldots d\mathbf{x}_n \tag{3.33b}$$

$$= \Pr\left(|\mathcal{X}| = 0\right) + \sum_{n=1}^{\infty} \Pr\left(|\mathcal{X}| = n\right) \int_{S \times \overset{n}{\ldots} \times S} f_{\mathcal{X},n}(\mathbf{x}_1, \ldots, \mathbf{x}_n)\, d\mathbf{x}_1 \ldots d\mathbf{x}_n \tag{3.33c}$$

$$= \Pr\left(|\mathcal{X}| = 0\right) + \sum_{n=1}^{\infty} \Pr\left(|\mathcal{X}| = n\right) \Pr\left(\mathbf{x}_1, \ldots, \mathbf{x}_n \in S || \mathcal{X}| = n \text{ with } \mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}\right). \tag{3.33d}$$

Finally, the total probability formula yields

$$\int_S f_{\mathcal{X}}(X)\, \delta X = \Pr\left(|\mathcal{X}| = 0\right) + \sum_{n=1}^{\infty} \Pr\left(|\mathcal{X}| = n\right) \Pr\left(\mathcal{X} \subset S || \mathcal{X}| = n\right) \tag{3.34a}$$

$$= \Pr\left(\mathcal{X} = \emptyset\right) + \Pr\left(\mathcal{X} \neq \emptyset \text{ and } \mathcal{X} \subset S\right) = \Pr\left(\mathcal{X} \subset S\right). \tag{3.34b}$$

□

3.6 Probability Generating Functionals

Probability generating functionals (PFGLs) are the RFS version of probability generating functions from standard probability theory, and have similar properties to them.

Before introducing the PGFL concept, let us stress the fact that PGFLs are functionals, i.e., they are functions that take in scalar valued functions of a vector variable as arguments and

return scalar values. This is in much contrast to moment generating functions, which are usual scalar valued functions of a scalar variable.

The concept of functional is an abstraction that has powerful applications not only to FISST, but also to other fields like Fourier Analysis, Partial Differential Equations, among others (see for example [25]). However, this abstraction might be difficult to grasp at the beginning since it revolves on the idea of a "function defined on a set of functions".

In order to illustrate the concept of a functional, let us consider one of the most important classes of functionals in Functional Analysis:

Definition 3.6.1. *(Scalar product functional) Let $g : \mathbb{R}^d \to \mathbb{R}$ be a fixed function. Then the scalar product functional T_g is defined as*

$$T_g(h) =< g, h >= \int_{\mathbb{R}^d} g(\mathbf{x})h(\mathbf{x})\, d\mathbf{x}\,, \tag{3.35}$$

where $h : \mathbb{R}^d \to \mathbb{R}$.

T_g is indeed a functional because it takes in functions $h : \mathbb{R}^d \to \mathbb{R}$ as arguments, which are called test functions, and returns scalars. However, note that $T_g(h)$ is not necessarily well-defined for any pair of functions $g, h : \mathbb{R}^d \to \mathbb{R}$. This is because the product gh may not be integrable on $\mathbb{R}^d$. The solution to this is to restrict the selection of functions g and h to sets of functions with "suitable properties", which ensure that the functional $T_g(h)$ is always well-defined.

The discussion of what is meant by "suitable properties" requires the study of measurable functions and Lebesgue spaces, among other concepts, and is therefore outside of the scope of this thesis. Moreover, such a measure theoretic discussion is unnecessary for the objectives of this thesis. The reason for this is that the derivation of the PMBM filter only requires to study the structure of the involved PGFLs (see section 4.4). In particular, the evaluation of these functionals on test functions is secondary.

For the applications of this thesis and for the sake of simplicity, we will only consider test functions that are continuous on $\mathbb{R}^d$, except on a discrete set of points, and that tend to zero when the argument $\mathbf{x} \in \mathbb{R}^d$ tends to infinity, i.e., $\lim_{|\mathbf{x}|\to\infty} h(\mathbf{x}) = 0$. Moreover, the set of all test functions with these properties is denoted by $\mathcal{C}(\mathbb{R}^d, \mathbb{R})$.

The set $\mathcal{C}(\mathbb{R}^d, \mathbb{R})$ is sufficiently "rich" with functions. In particular, it contains all continuous probability density functions and all Dirac deltas.

For an introduction to measure theory, we refer to [9].

Definition 3.6.2 (Probability Generating Functional)**.** *Let $\mathcal{X}$ be RFS on $\mathbb{R}^d$ with set density function $f_{\mathcal{X}}$. Then its associated probability generating functional (PGFL) is defined as*

$$F_{\mathcal{X}}(h) = \int h^X f_{\mathcal{X}}(X)\, \delta X\,, \tag{3.36}$$

where $h \in \mathcal{C}(\mathbb{R}^d, \mathbb{R})$ is a test function, and the notation h^X denotes the finite product $\prod_{\mathbf{x}\in X} h(\mathbf{x})$ with the convention that $h^{\emptyset} = 1$.

Similarly to probability generating functions in standard probability theory, the PGFL of the union of a finite number of independent RFSs is the product of the corresponding PGFL, as shown in the following result:

Theorem 3.6.1. *(PGFLs and independence) Let $\mathcal{X}_1, \ldots, \mathcal{X}_n$ be a finite number of independent RFSs with the same base space $\mathbb{R}^d$.*

Then the PGFL of the RFS $\mathcal{X}$ defined as $\mathcal{X} = \cup_{i=1}^{m} \mathcal{X}_i$ is

$$F_{\mathcal{X}} = \prod_{i=1}^{n} F_{\mathcal{X}_i}. \tag{3.37}$$

Proof. Let h be a test function. Then by the definition of the PGFL and the set integral (Definition 3.6.2 and Definition 3.5.1), we have that

$$\prod_{i=1}^{n} F_{\mathcal{X}_i}(h) \tag{3.38a}$$

$$= \prod_{i=1}^{n} \left(f_{\mathcal{X}_i}(\emptyset) + \sum_{m=1}^{\infty} \frac{1}{m!} \int \left(\prod_{j=1}^{m} h(\mathbf{x}_j) \right) f_{\mathcal{X}_i}(\{\mathbf{x}_1, \ldots, \mathbf{x}_m)\})\, \delta\{\mathbf{x}_1, \ldots, \mathbf{x}_m)\} \right). \tag{3.38b}$$

The application of Proposition 3.2.1 on this expression yields

$$\prod_{i=1}^{n} F_{\mathcal{X}_i}(h) \tag{3.39a}$$

$$= \prod_{i=1}^{n} \left(f_{\mathcal{X}_i}(\emptyset) + \sum_{m=1}^{\infty} \Pr(|\mathcal{X}_i| = m) \int \left(\prod_{j=1}^{m} h(\mathbf{x}_j) \right) f_{\mathcal{X}_i,m}(\mathbf{x}_1, \ldots, \mathbf{x}_m))\, d\mathbf{x}_1 \ldots d\mathbf{x}_m \right). \tag{3.39b}$$

By multiplying out all factors in (3.39b), and by grouping the terms whose sum of integral dimensions is the same, we have that

$$\prod_{i=1}^{n} F_{\mathcal{X}_i}(h) = \prod_{i=1}^{n} f_{\mathcal{X}_i}(\emptyset) + \ldots \tag{3.40a}$$

$$\ldots \sum_{m=1}^{\infty} \sum_{\substack{0\le 1\le m_i \\ \sum m_i = m}} \frac{m!}{\prod_{i=1}^{n} m_i!} \prod_{i=1}^{n} \left(\Pr(|\mathcal{X}_i| = m_i) \int \left(\prod_{l=1}^{m_i} h(\mathbf{x}_l) \right) f_{\mathcal{X}_i,m_i}(\mathbf{x}_1, \ldots, \mathbf{x}_{m_i}))\, d\mathbf{x}_1 \ldots d\mathbf{x}_{m_i} \right), \tag{3.40b}$$

where the term $\frac{m!}{\prod_{i=1}^{n} m_i!}$ is the number of different ways one can split a set of m elements into n subsets $X_1, \ldots, X_n$, where the number of elements in X_i is m_i.

By grouping all the integrands of each term in (3.40b) under the same integral, and by using that the marginal probability density functions $f_{\mathcal{X}_\rangle,m_i}$ correspond to independent distribution, we conclude that

$$\prod_{i=1}^{n} F_{\mathcal{X}_i}(h) = \prod_{i=1}^{n} f_{\mathcal{X}_\rangle}(\emptyset) + \dots \tag{3.41a}$$

$$\cdots \sum_{m=1}^{\infty} \left(\sum_{\substack{0\leq 1\leq m_i \\ \sum m_i = m}} \frac{m!}{\prod_{i=1}^{n} m_i!} \prod_{i=1}^{n} \left(\Pr\left(|\mathcal{X}_i| = m_i\right)\right) \right) \int \left(\prod_{i=1}^{n} h(\mathbf{x}_i) \right) f_{\mathcal{X},m}(\mathbf{x}_1, \dots, \mathbf{x}_m))\, d\mathbf{x}_1 \dots d\mathbf{x}_m. \tag{3.41b}$$

Now, we observe that

$$\sum_{\substack{0\leq 1\leq m_i \\ \sum m_i = m}} \frac{m!}{\prod_{i=1}^{n} m_i!} \prod_{i=1}^{n} \left(\Pr\left(|\mathcal{X}_i| = m_i\right)\right) = \Pr\left(|\mathcal{X}| = m\right) \tag{3.42}$$

since the left-hand side of this expression is the sum over all ways one can split a set of m elements into n subsets $X_1, \dots, X_n$, where the number of elements in X_i is m_i multiplied by the probabilities that these sets have m_i elements.

Hence, by inserting of (3.42) into (3.41b), it follows that

$$\prod_{i=1}^{n} F_{\mathcal{X}_i}(h) = \prod_{i=1}^{n} f_{\mathcal{X}_\rangle}(\emptyset) + \dots \tag{3.43a}$$

$$\cdots \sum_{m=1}^{\infty} \Pr\left(|\mathcal{X}| = m\right) \int \left(\prod_{i=1}^{n} h(\mathbf{x}_i) \right) f_{\mathcal{X},m}(\mathbf{x}_1, \dots, \mathbf{x}_m))\, d\mathbf{x}_1 \dots d\mathbf{x}_m \tag{3.43b}$$

$$= \prod_{i=1}^{n} \Pr\left(\mathcal{X}_i = \emptyset\right) + \sum_{m=1}^{\infty} \Pr\left(|\mathcal{X}| = m\right) \int \left(\prod_{i=1}^{n} h(\mathbf{x}_i) \right) f_{\mathcal{X},m}(\mathbf{x}_1, \dots, \mathbf{x}_m))\, d\mathbf{x}_1 \dots d\mathbf{x}_m \tag{3.43c}$$

$$= \Pr\left(\mathcal{X} = \emptyset\right) + \sum_{m=1}^{\infty} \frac{1}{m!} \int \left(\prod_{j=1}^{m} h(\mathbf{x}_j) \right) f_{\mathcal{X}}(\{\mathbf{x}_1, \dots, \mathbf{x}_m)\})\, \delta\{\mathbf{x}_1, \dots, \mathbf{x}_m)\} \tag{3.43d}$$

$$= F_{\mathcal{X}}(h). \tag{3.43e}$$

□

By using Definition 3.6.2 together with Theorem 3.6.1, the PGFL of a wide variaty of RFSs can be found based on the PGFL expressions for RFS with simpler set density functions. In particular, the PFGLs of the important classes of RFSs presented in Section 3.4 are given by:

Proposition 3.6.1. *(The PGFL of a PPP) Let $\mathcal{X}$ be a PPP with intensity function $D(\mathbf{x})$. Then its PGFL is given by*

$$F_{\mathcal{X}}(h) = e^{-<D,h-1>} \tag{3.44}$$

Proof. By inserting (3.20) into the definition of the PGFL (3.36), and using the definition of the set integral (3.31), it follows that:

$$F_{\mathcal{X}}(h) = \int h^X f_{\mathcal{X}}(X)\,\delta X = \int \prod_{\mathbf{x}\in X} h(\mathbf{x}) e^{-<D,1>} \prod_{\mathbf{x}\in X} D(\mathbf{x})\,\delta X \tag{3.45a}$$

$$= e^{-<D,1>}\left(1 + \sum_{n=1}^{\infty} \frac{1}{n!} \int \prod_{i=1}^{n} (h(\mathbf{x}_i) D(\mathbf{x}_i))\, d\mathbf{x}_1 \ldots d\mathbf{x}_n\right). \tag{3.45b}$$

The final expression for the PGFL of a PPP $\mathcal{X}$ is then obtained by exchanging the order of the product and the integration. This yields:

$$F_{\mathcal{X}}(h) = e^{-<D,1>}\left(1 + \sum_{n=1}^{\infty} \frac{1}{n!} \prod_{i=1}^{n} \left(\int h(\mathbf{x}_i) D(\mathbf{x}_i)\, d\mathbf{x}_i\right)\right) \tag{3.46a}$$

$$= e^{-<D,1>}\left(1 + \sum_{n=1}^{\infty} \frac{1}{n!} \prod_{i=1}^{n} < h, D >\right) \tag{3.46b}$$

$$= e^{-<D,1>}\left(1 + \sum_{n=1}^{\infty} \frac{1}{n!} < h, D >^n\right) \tag{3.46c}$$

$$= e^{-<D,1>} e^{<h,D>} = e^{<D,h-1>}. \tag{3.46d}$$

□

Proposition 3.6.2. *(The PGFL of a Bernoulli process) Let $\mathcal{X}$ be a Bernoulli process with existence probability r and distribution f. Then its PGFL is given by*

$$F_{\mathcal{X}}(h) = 1 - r + r < f, h > . \tag{3.47}$$

Proof. This result follows directly from the expression for the set density of a Bernoulli process (3.22) and the definitions of the PGFL (3.36) and the set integral (3.31):

$$F_{\mathcal{X}}(h) = \int h^X f_{\mathcal{X}}(X)\,\delta X \tag{3.48a}$$

$$= f_{\mathcal{X}}(\emptyset) + \int h(\mathbf{x}) r f(\mathbf{x})\, dx + 0 + 0 + \ldots \tag{3.48b}$$

$$= 1 - r + r < f, h > . \tag{3.48c}$$

□

Proposition 3.6.3. *(The PGFL of a MB) Let $\mathcal{X}$ be a MB processes with parameters $\{(r^i, f^i)\}_{i\in\mathbb{I}}$. Then its PGFL is given by*

$$F_{\mathcal{X}}(h) = \prod_{i\in\mathbb{I}} \left(1 - r^i + r^i < f^i, h >\right). \tag{3.49}$$

Proof. By definition (see Section 3.4.3), the MB process $\mathcal{X}$ is the union of a finite number of independent Bernoulli processes $\mathcal{X}_i$, where each $\mathcal{X}_i$ has existence probability r^i and distribution f^i. Hence, the result we want to prove, follows directly from Theorem 3.6.1 and the previous proposition (Proposition 3.6.2). □

Proposition 3.6.4. *(The PGFL of a MBM) Let $\mathcal{X}$ be a MBM with parameters*

$$\{w^j, \{(r^{j,i}, f^{j,i})\}_{i\in\mathbb{I}^j}\}_{j\in\mathbb{J}}. \tag{3.50}$$

Then its PGFL is given by

$$F_{\mathcal{X}}(h) = \sum_{j\in\mathbb{J}} w^j \prod_{i\in\mathbb{I}^j} \left(1 - r^{j,i} + r^{j,i} < f^{j,i}, h > \right). \tag{3.51}$$

Proof. By definition (see Section 3.4.4), the set density function of the MBM process $\mathcal{X}$ is

$$f_{\mathcal{X}}(X) = \sum_{j\in\mathbb{J}} w^j f_{\mathcal{X}_j}(X), \tag{3.52}$$

where $f_{\mathcal{X}_j}(X)$ is the set density function of a MB process with parameters $\{(r^{j,i}, f^{j,i})\}_{i\in\mathbb{I}^j}$.

Hence, by setting the expression for $f_{\mathcal{X}}$ (3.52) into the PGFL definition (3.36), and using the linearity of the set integral (3.31) together with the previous proposition (Equation (3.51)), the result follows:

$$\begin{aligned} F_{\mathcal{X}}(h) &= \int h^X f_{\mathcal{X}}(X)\,\delta X && \text{(3.53a)}\\ &= \int h^X \sum_{j\in\mathbb{J}} w^j f_{\mathcal{X}_j}(X)\,\delta X && \text{(3.53b)}\\ &= \sum_{j\in\mathbb{J}} w^j \int h^X f_{\mathcal{X}_j}(X)\,\delta X && \text{(3.53c)}\\ &= \sum_{j\in\mathbb{J}} w^j F_{\mathcal{X}_i}(h) = \sum_{j\in\mathbb{J}} w^j \prod_{i\in\mathbb{I}^j} \left(1 - r^{j,i} + r^{j,i} < f^{j,i}, h > \right). && \text{(3.53d)} \end{aligned}$$

□

From propositions 3.6.1-3.51, we observe that the function scalar product $< \cdot, \cdot >$ is a central building block for the PGFLs of PPPs, Bernouli processes, MBs and MBMs.

Another important result on PGFLs, it that a PGFL contains all the statistical information of its corresponding RFS because the set density function can be retrieved from this PGFL by derivation. Before presenting this result, let us define how functionals are derivated:

Definition 3.6.3. *(Functional derivative) Let $F_{\mathcal{X}}$ be the PGFL of the RFS $\mathcal{X}$. The functional derivative of $F_{\mathcal{X}}$ with respect to a finite and non-empty set $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ is given by the iterated derivative*

$$\frac{\delta F_{\mathcal{X}}}{\delta X} = \frac{\delta^n F_{\mathcal{X}}}{\delta \mathbf{x}_1 \ldots \delta \mathbf{x}_n}, \tag{3.54}$$

where the individual derivatives are the limit values

$$\frac{\delta F_{\mathcal{X}}}{\delta \mathbf{x}_i}(h) = \lim_{\epsilon\to 0^+} \frac{\delta F_{\mathcal{X}}[h + \epsilon\delta_{\mathbf{x}_i}] - \delta F_{\mathcal{X}}(h)}{\epsilon} \tag{3.55}$$

and $\delta_{\mathbf{x}_i}(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{x}_i)$ is the Dirac delta at the point $x_i \in \mathbb{R}^d$.

Furthermore, functional derivative of $F_{\mathcal{X}}$ with respect to the empty set is itself, i.e.,

$$\frac{\delta F_{\mathcal{X}}}{\delta \emptyset} = F_{\mathcal{X}}. \tag{3.56}$$

Theorem 3.6.2. *(Retrieving set density from PGFL) Let $F_{\mathcal{X}}$ be the PGFL of a RFS$\mathcal{X}$. Then the set density function $f_{\mathcal{X}}$ can be recovered from the PGFL according to the formula*

$$f_{\mathcal{X}}(X) = \frac{\delta F_{\mathcal{X}}}{\delta X}[0]. \tag{3.57}$$

Proof. First, if $X = \emptyset$, then the functional derivative with respect to X leaves the functional. Hence, it follows from the definition of the PGFL (Definition 3.6.2) that

$$\frac{\delta F_{\mathcal{X}}}{\delta X}[0] = F_{\mathcal{X}}(h)|_{h=0} = \int h^X f_{\mathcal{X}}(X)\, \delta X \Big|_{h=0} \tag{3.58a}$$

$$= f_{\mathcal{X}}(\emptyset) + \sum_{m=1}^{\infty} \frac{1}{m!} \int \left(\prod_{j=n+1}^{m} h(\mathbf{x}_j) \right) f_{\mathcal{X}}(\{\mathbf{x}_1, \ldots, \mathbf{x}_m)\})\, \delta\{\mathbf{x}_1, \ldots, \mathbf{x}_m)\} \Big|_{h=0} \tag{3.58b}$$

$$= f_{\mathcal{X}}(\emptyset) = f_{\mathcal{X}}(X). \tag{3.58c}$$

Let us now prove the identity for a non-empty set $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ with $|X| = n$. In order to do so, consider m different vectors $x_1, \ldots, x_m$. Then, by multiplying out the terms in $\prod_{i=n+1}^{m} \left(h(\mathbf{x}_i) + \epsilon \delta_{\mathbf{x}_{n+1}}(\mathbf{x}_i)\right)$, we have that

$$\lim_{\epsilon \to 0^+} \int \left(\frac{\prod_{i=n+1}^{m} \left(h(\mathbf{x}_i) + \epsilon \delta_{\mathbf{x}_{n+1}}(\mathbf{x}_i)\right) - \prod_{i=n+1}^{m} h(\mathbf{x}_i)}{\epsilon} \right) f_{\mathcal{X},m}(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{x}_{n+1}, \ldots \mathbf{x}_m)\, d\mathbf{x}_{n+1} \ldots d\mathbf{x}_m \tag{3.59a}$$

$$= \lim_{\epsilon \to 0^+} \int \left(\frac{\epsilon \left(\sum_{j=n+1}^{m} \delta_{\mathbf{x}_{n+1}}(\mathbf{x}_j) \prod_{\substack{i=n+1 \\ i \neq j}}^{m} h(\mathbf{x}_i) \right) + \epsilon^2(\ldots)}{\epsilon} \right) f_{\mathcal{X},m}(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{x}_{n+1}, \ldots \mathbf{x}_m)\, d\mathbf{x}_{n+1} \ldots d\mathbf{x}_m \tag{3.59b}$$

where the terms that have a second or higher powers of ϵ are factorized as $\epsilon^2(\ldots)$. These terms do not contribute to the limit since $\epsilon \to 0+$, and can be eliminated. Hence, by dividing by ϵ and taking the limit, we have that

$$\lim_{\epsilon \to 0^+} \int \left(\frac{\prod_{i=n+1}^{m} \left(h(\mathbf{x}_i) + \epsilon \delta_{\mathbf{x}_{n+1}}(\mathbf{x}_i)\right) - \prod_{i=n+1}^{m} h(\mathbf{x}_i)}{\epsilon} \right) f_{\mathcal{X},m}(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{x}_{n+1}, \ldots \mathbf{x}_m)\, d\mathbf{x}_{n+1} \ldots d\mathbf{x}_m \tag{3.60a}$$

$$= \int \left(\sum_{j=n+1}^{m} \delta_{\mathbf{x}_{n+1}}(\mathbf{x}_j) \prod_{\substack{i=n+1 \\ i \neq j}}^{m} h(\mathbf{x}_i) \right) f_{\mathcal{X},m}(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{x}_{n+1}, \ldots \mathbf{x}_m)\, d\mathbf{x}_{n+1} \ldots d\mathbf{x}_m\,, \tag{3.60b}$$

By the linearity of the integral, and the fact that $\mathbf{x}1, \ldots, \mathbf{x}_n$ are not integration variables, it follows that

$$\lim_{\epsilon \to 0^+} \int \left(\frac{\prod_{i=n+1}^{m} \left(h(\mathbf{x}_i) + \epsilon \delta_{\mathbf{x}_{n+1}}(\mathbf{x}_i) \right) - \prod_{i=n+1}^{m} h(\mathbf{x}_i)}{\epsilon} \right) f_{\mathcal{X},m}(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{x}_{n+1}, \ldots \mathbf{x}_m) \, d\mathbf{x}_{n+1} \ldots d\mathbf{x}_m \tag{3.61a}$$

$$= \sum_{j=n+1}^{m} \left(\prod_{\substack{i=n+1 \\ i \neq j}}^{m} h(\mathbf{x}_i) \right) \int \delta_{\mathbf{x}_{n+1}}(\mathbf{x}_j) f_{\mathcal{X},m}(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{x}_{n+1}, \ldots \mathbf{x}_m) \, d\mathbf{x}_{n+1} \ldots d\mathbf{x}_m \,, \tag{3.61b}$$

Since the function $f_{\mathcal{X},m}$ is invariant under permutations of its variables, we can always interchange x_j with x_{n+1} in the integrals. By doing so, the integral of the product between $\delta_{\mathbf{x}_{n+1}}(\mathbf{x}_j)$ and $f_{\mathcal{X},m}$ with respect to $\mathbf{x}_j$, will fix the variable at the position $n+1$ to x_{n+1} in $f_{\mathcal{X},m}$ Therefore, the limit becomes

$$\lim_{\epsilon \to 0^+} \int \left(\frac{\prod_{i=n+1}^{m} \left(h(\mathbf{x}_i) + \epsilon \delta_{\mathbf{x}_{n+1}}(\mathbf{x}_i) \right) - \prod_{i=n+1}^{m} h(\mathbf{x}_i)}{\epsilon} \right) f_{\mathcal{X},m}(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{x}_{n+1}, \ldots \mathbf{x}_m) \, d\mathbf{x}_{n+1} \ldots d\mathbf{x}_m \tag{3.62a}$$

$$= (m-n) \left(\prod_{\substack{i=n+1 \\ i \neq j}}^{m} h(\mathbf{x}_i) \right) \int f_{\mathcal{X},m}(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{x}_{n+1}, \ldots \mathbf{x}_m) \, d\mathbf{x}_{n+2} \ldots d\mathbf{x}_m \,, \tag{3.62b}$$

if $n+1 < m$, or

$$\lim_{\epsilon \to 0^+} \int \left(\frac{\prod_{i=n+1}^{m} \left(h(\mathbf{x}_i) + \epsilon \delta_{\mathbf{x}_{n+1}}(\mathbf{x}_i) \right) - \prod_{i=n+1}^{m} h(\mathbf{x}_i)}{\epsilon} \right) f_{\mathcal{X},m}(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{x}_{n+1}, \ldots \mathbf{x}_m) \, d\mathbf{x}_{n+1} \ldots d\mathbf{x}_m \tag{3.63a}$$

$$= (m-n) \left(\prod_{\substack{i=n+1 \\ i \neq j}}^{m} h(\mathbf{x}_i) \right) f_{\mathcal{X},m}(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{x}_{n+1}, \ldots \mathbf{x}_m) \,, \tag{3.63b}$$

if $n+1 = m$.

Now, since the PGFL $F_{\mathcal{X}}$ is defined as

$$F_{\mathcal{X}}(h) = f_{\mathcal{X}}(\emptyset) + \sum_{m=1}^{\infty} \frac{1}{m!} \int \left(\prod_{j=n+1}^{m} h(\mathbf{x}_j) \right) f_{\mathcal{X}}(\{\mathbf{x}_1, \ldots, \mathbf{x}_m)\}) \, \delta\{\mathbf{x}_1, \ldots, \mathbf{x}_m)\} \,, \tag{3.64}$$

we can use the expressions in (3.63b) and (3.63b) to prove by induction over the number of iterated functional derivatives with respect to vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$ that

$$\frac{\delta^n F_{\mathcal{X}}}{\delta \mathbf{x}_1 \ldots \delta \mathbf{x}_n}(h) = f_{\mathcal{X},n}(\mathbf{x}_1, \ldots, \mathbf{x}_n) + \ldots \tag{3.65a}$$

$$\cdots \sum_{m=n+1}^{\infty} \frac{1}{(m-n)!} \left(\prod_{i=n+1}^{m} h(\mathbf{x}_i) \right) \int \left(\prod_{i=n+1}^{m} h(\mathbf{x}_i) \right) f_{\mathcal{X},m}(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{x}_{n+1}, \ldots \mathbf{x}_m) \, d\mathbf{x}_{n+1} \ldots d\mathbf{x}_m. \tag{3.65b}$$

In particular, evaluation of this expression at $h = 0$ gives the final result:

$$\frac{\delta^n F_{\mathcal{X}}}{\delta \mathbf{x}_1 \dots \delta \mathbf{x}_n}(h) = f_{\mathcal{X},n}(\mathbf{x}_1, \dots, \mathbf{x}_n) \tag{3.66a}$$

$$= f_{\mathcal{X}}(\{(\mathbf{x}_1, \dots, \mathbf{x}_n)\}). \tag{3.66b}$$

□

Note that the functional derivative of a PGFL (3.54)-(3.55) is another functional. Moreover, it is immediate to verify that the functional derivative is a linear operator in the sense that

$$\frac{\delta(\alpha F_{\mathcal{X}} + \beta F_{\mathcal{Y}})}{\delta \mathbf{x}}(h) = \alpha \frac{\delta F_{\mathcal{X}}}{\delta \mathbf{x}}(h) + \beta \frac{\delta F_{\mathcal{Y}}}{\delta \mathbf{x}}(h) \tag{3.67}$$

for any scalars $\alpha, \beta \in \mathbb{R}$ and any PFGLs $\mathcal{X}$ and $\mathcal{Y}$ with the same base space $\mathbb{R}^d$.

There are many other "turn-the-crank" derivation rules for functional derivatives similar to the ones in classical Calculus. However, for the derivation of the PMBM filter (see Section 4.4) only one additional rule is necessary: The expression for the functional derivative of the union of a finite number of independent RFSs.

Theorem 3.6.3. *(Product rule for functionals) Let $F_1, \dots, F_n$ be a finite number of functionals for test functions defined on $\mathbb{R}^d$.*

Then the functional derivative of their product is given by

$$\frac{\delta \prod_{i=1}^n F_i}{\delta X} = \sum_{\uplus_{i=1}^n X_i = X} \prod_{i=1}^n \frac{\delta F_i}{\delta X_i}. \tag{3.68}$$

Proof. First, if $X = \emptyset$, then the functional derivative with respect to X leaves the functional unchanged. In addition, the only partition of $X = \emptyset$ into sets X_i is $X_i = \emptyset$ for all i. Hence,

$$\frac{\delta \prod_{i=1}^n F_i}{\delta X} = \prod_{i=1}^n F_i = \sum_{\uplus_{i=1}^n X_i = X} \prod_{i=1}^n \delta F_i = \sum_{\uplus_{i=1}^n X_i = X} \prod_{i=1}^n \frac{\delta F_i}{\delta X_i}. \tag{3.69}$$

Consider now a non-empty set $X = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ with $|X| = n$. Then, we prove the product rule identity by induction over the cardinality of X:

Let $n = 1$, i.e., $X = \{\mathbf{x}\}$. For only a vector $\mathbf{x}$, it follows from the definition of the functional derivative with respect to a vector (Definition 3.6.3) and the standard product rule from calculus that

$$\frac{\delta \prod_{i=1}^n F_i}{\delta X} = \frac{\delta \prod_{i=1}^n F_i}{\delta \mathbf{x}} = \sum_{j=1}^n \frac{\delta F_j}{\delta \mathbf{x}} \prod_{\substack{i=1 \\ i \neq j}}^n F_i \tag{3.70a}$$

$$= \sum_{j=1}^n \frac{\delta F_j}{\delta \mathbf{x}} \prod_{\substack{i=1 \\ i \neq j}}^n \frac{\delta F_j}{\delta \emptyset} \tag{3.70b}$$

$$= \sum_{\uplus_{i=1}^n X_i = X} \prod_{i=1}^n \frac{\delta F_i}{\delta X_i}, \tag{3.70c}$$

where the last equality follows from the fact that the only way to partition the set $X = \{\mathbf{x}\}$ is if one and only one set X_i is equal to X and the rest are empty sets.

Assume now that identity is true for sets with n elements, and consider a set with $n+1$ elements X. Hence, $X = Y \cup \{\mathbf{x}\}$, where Y is a set with n elements. By using the definition of the functional derivative with respect to a vector (Definition 3.6.3), the inductions hypothesis and the identity (3.70b), we have that

$$\frac{\delta \prod_{i=1}^{n} F_i}{\delta X} = \frac{\delta}{\delta x} \frac{\delta \prod_{i=1}^{n} F_i}{\delta Y} \tag{3.71a}$$

$$= \frac{\delta}{\delta x} \left(\sum_{\biguplus_{i=1}^{n} Y_i = Y} \prod_{i=1}^{n} \frac{\delta F_i}{\delta Y_i} \right) \tag{3.71b}$$

$$= \sum_{\biguplus_{i=1}^{n} Y_i = Y} \frac{\delta}{\delta x} \prod_{i=1}^{n} \frac{\delta F_i}{\delta Y_i} \tag{3.71c}$$

$$= \sum_{\biguplus_{i=1}^{n} Y_i = Y} \sum_{j=1}^{n} \frac{\delta}{\delta \mathbf{x}} \frac{\delta F_j}{\delta Y_j} \prod_{\substack{i=1 \\ i \neq j}}^{n} \frac{\delta}{\delta \emptyset} \frac{\delta F_i}{\delta Y_i} \tag{3.71d}$$

$$= \sum_{\biguplus_{i=1}^{n} Y_i = Y} \sum_{j=1}^{n} \frac{\delta F_j}{\delta \{\mathbf{x}\} \cup Y_j} \prod_{\substack{i=1 \\ i \neq j}}^{n} \frac{\delta F_i}{\delta Y_i} \tag{3.71e}$$

$$= \sum_{\biguplus_{i=1}^{n} X_i = X} \prod_{i=1}^{n} \frac{\delta F_i}{\delta X_i}\,, \tag{3.71f}$$

where the last equality follows from the fact that any partition of X into $X_1, \ldots, X_n$ is given by a partition of Y into $Y_1, \ldots, Y_n$, where $X_i = Y_i$ for all i, except for one index j that verifies $X_j = Y_j \cup \{\mathbf{x}\}$. □

Chapter 4

Poisson multi-Bernoulli mixture filter for multiple extended object tracking

The Poisson multi-Bernoulli mixture (PMBM) filter for MEOT is one of the most sophisticated and state-of-the-art MEOT methods. The theory behind this filter is based on the concepts and results about RFSs that have been presented in Chapter 3. Because of the considerable theoretical prerequisites needed for this filter, papers that present this filter do not give a detailed presentation of it, and they usually refer to other sources for some details.

In this chapter, we present a self-contained derivation of the PMBM filter that is accessible from an undergraduate level in mathematics and statistics.

4.1 The dynamic and measurement models of the object

The information about the object's movement and extent is encapsulated in its corresponding state vector $\mathbf{x}$. The objective of the PMBM filter for MEOT is then to discover all existing objects and provide estimates of their respective state vectors over time. This is achieved by using the measurements $\mathbf{z}$ that are available, as well as stochastic models for both the object's dynamics and the measurements.

These stochastic models are discrete in time. In particular, the state vector values can only vary at discrete time points. In addition, at each discrete time point, new objects can be generated and existing objects can cease to exist. Furthermore, the measurements available at a time point can be generated by the existing objects, but can also be due to ”false alarms”, which are also known as ”clutter”.

The detailed formal description of these models is presented below.

4.1.1 Description of the dynamic model:

A.1 The total number of objects at each time point is finite.

A.2 New objects generate at each time point according to a PPP with birth intensity $D^b(\mathbf{x})$.

A.3 An existing object survives to the next time point with probability $p^s(\mathbf{x})$. This survival process is independent and identically distributed for each existing object.

A.4 If the existing object survived, its state vector at a time point, $\mathbf{x}_{k-1}$, becomes $\mathbf{x}_k$ at the next time point according to the transition probability density function $f_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$. This transition process is independent and identically distributed for each existing object.

A.5 The birth of new targets and the joint process of survival and state-transition for each target are independent processes.

4.1.2 Description of the measurement model:

B.1 The number of measurements at each time point is finite.

B.2 Clutter or false alarms are generated according to a PPP with intensity $D^c(\mathbf{x})$, which is independent of the objects and object-generated measurements.

B.3 An existing object can be detected with detection probability $p^d(\mathbf{x})$.

B.4 If an existing object is detected, it generates measurements according to a PPP with intensity $\lambda^m(\mathbf{x})l(\mathbf{z}|\mathbf{x})$, where $\lambda^m(\mathbf{x})$ is the Poisson rate and $l(\mathbf{z}|\mathbf{x})$ is the spatial distribution of the PPP.

B.5 A measurement is either a false alarm or it is generated by one and only one object.

B.6 A measurement generated by an object is independent of all other objects and their object-generated measurements. Note that measurements generated by the same object can be dependent.

Note that although an object is "detected" according to (**B.3**), it can still generate zero measurements since the cardinality of the measurement set is Poisson distributed according to (**B.4**). Therefore, the effective probability of detection is

$$p^d(\mathbf{x})\left(1 - e^{-\lambda^m(\mathbf{x})}\right), \tag{4.1}$$

where $1 - e^{-\lambda^m(\mathbf{x})}$ is the Poisson probability of generating at least one measurement. Hence, the effective probability of missed detection is

$$q^d(\mathbf{x}) = 1 - p^d(\mathbf{x})\left(1 - e^{-\lambda^m(\mathbf{x})}\right) = 1 - p^d(\mathbf{x}) + p^d(\mathbf{x})e^{-\lambda^m(\mathbf{x})}. \tag{4.2}$$

Furthermore, the probability density function $l(\mathbf{z}|\mathbf{x})$ is the likelihood of the measurement $\mathbf{z}$ given the state vector $\mathbf{x}$. Therefore, the likelihood of a non-empty set of measurements Z given a state vector $\mathbf{x}$ is given by the product of the probability of detection and the PPP density for the set Z, i.e.,

$$l_Z(\mathbf{x}) = p^d(\mathbf{x}) \prod_{\mathbf{z}\in Z} \lambda^m(\mathbf{x}) l(\mathbf{z}|\mathbf{x}), \tag{4.3}$$

while the likelihood of an empty set of measurements given a state vector $\mathbf{x}$ is

$$l_{\emptyset}(\mathbf{x}) = q^d(\mathbf{x}) \tag{4.4}$$

4.2 The data association problem

According to **B.5**, any measurement is either clutter or generated by one and only one object. However, measurements do not provide a priori information about their origin. In addition, among the object-generated measurements, it is unknown which measurements originated from objects that have been previously detected ("tracked"), and which come from undetected ("unknown") objects. Since the measurement origin is unknown, it is necessary to discuss data association, which is the problem of generating and bookkeeping hypotheses that associate measurements to particular tracked objects or to the "background", which is the common term for both clutter and unknown objects.

Let the set of all measurements for a given time step Z be indexed by the set $\mathbb{M}$, i.e.,

$$Z = \{\mathbf{z}_m\}_{m \in \mathbb{M}}\,, \tag{4.5}$$

and let $\mathbb{I}$ be the index set of the tracked objects with $\mathbb{M} \cap \mathbb{I} = \emptyset$.

If both $\mathbb{M}$ and $\mathbb{I}$ are empty, then there are no measurements and no tracked objects. In particular, there is no data association problem for this time point.

Therefore, let us assume that $\mathbb{M}$ or $\mathbb{I}$ are non-empty. Then, a data association hypothesis A is any partition of $\mathbb{M} \cup \mathbb{I}$ into non-empty sets $C_1, \ldots, C_{N_A}$, known as cells, that contain at most one element from $\mathbb{I}$. In other words, the cells $C_1, \ldots, C_{N_A}$ are characterized by the conditions

$$\biguplus_{n=1}^{N_A} C_n = \mathbb{M} \cup \mathbb{I} \tag{4.6a}$$

$$C_n \neq \emptyset \tag{4.6b}$$

$$|C_n \cap \mathbb{I}| \leq 1. \tag{4.6c}$$

Moreover, we write $A = \{C_1, C_1, \ldots, C_{N_A}\}$.

Furthermore, let Z_C denote the set of measurements that corresponds to a cell C, i.e.,

$$Z_C = \bigcup_{m \in C \cap \mathbb{M}} \{z_m\}\,, \tag{4.7}$$

and let $l_C(\mathbf{x})$ be the likelihood of the cell measurements Z_C given the state vector $\mathbf{x}$, i.e.,

$$l_C(\mathbf{x}) = l_{Z_C}(\mathbf{x}). \tag{4.8}$$

The interpretation of a cell C_n of the data association hypothesis A can be summarized in three possibilities:

1. If $C_n \cap \mathbb{I} \neq \emptyset$, then necessarily $C_n \cap \mathbb{I} = \{i\}$ for some index $i \in \mathbb{I}$. In such case, the measurements with indices in $C_n \cap \mathbb{M}$ are associated to the tracked target with index i.

2. If $C_n \cap \mathbb{I} = \emptyset$, then the measurements with indices in $C_n \cap \mathbb{M}$ are associated to the background, i.e., they are clutter or they are generated by unknown targets.

3. If $C_n \cap \mathbb{I} = \{i\}$ and $C_n \cap \mathbb{M} = \emptyset$, then the tracked target with index i has not been detected because it does not have associated measurements.

Data association example

In order to illustrate the concept of data association hypothesis and recognize the complexity of the data association problem, let us consider the simple example given in fig. 4.1. Here, two objects are been tracked, and their respective state vector estimates $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$ are assumed to be normally distributed. The ellipses in the figure represent the respective mean values and covariance matrices of these estimates. Moreover, the measurement model in this example gives direct measurements of the state vector, i.e., $\mathbf{z} = \mathbf{x} + \mathbf{w}$, where $\mathbf{w}$ is white Gaussian noise. There are three available measurements: $\mathbf{z}_1$, $\mathbf{z}_2$ and $\mathbf{z}_3$.

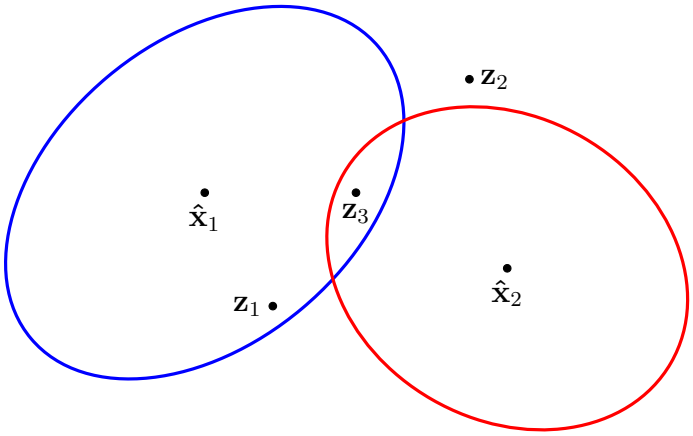


Figure 4.1: Data association example. Two tracked objects with normally distributed state vector estimates $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$, respectively. The ellipses represent their respective mean values and covariance matrices. The measurement model is $\mathbf{z} = \mathbf{x} + \mathbf{w}$, where $\mathbf{w}$ is white Gaussian noise. The available measurements are $\mathbf{z}_1$, $\mathbf{z}_2$ and $\mathbf{z}_3$.

Let $\mathbb{M} = \{m_1, m_2, m_3\}$ and $\mathbb{I} = \{i_1, i_2\}$ be index sets for the measurements and the tracked objects, respectively, such that $\mathbf{z}_{m_n} = \mathbf{z}_n$ and $\mathbf{x}_{i_n} = \mathbf{x}_n$. Then, one possible data association hypothesis is

$$C_1 = \{m_1, m_3, i_1\}, \quad C_2 = \{m_2\}, \quad C_3 = \{i_2\}, \tag{4.9}$$

which means that the measurements $\mathbf{z}_1$ and $\mathbf{z}_3$ are associated to the tracked object with state $\mathbf{x}_1$, the measurement $\mathbf{z}_2$ is clutter or is generated by an unknown object, and that the other tracked object has not been detected. Moreover, it follows that

$$Z_{C_1} = \{\mathbf{z}_1, \mathbf{z}_3\}, \quad Z_{C_2} = \{\mathbf{z}_2\}, \quad Z_{C_3} = \emptyset. \tag{4.10}$$

It is easy to verify that there are many other possible data association hypotheses. Moreover, some of these data association hypotheses are more likely than others. Nevertheless, one can not know a priori which of the data association hypotheses is the correct one.

4.3 The Poisson and multi-Bernoulli mixture model

The strategy followed by the PMBM filter to deal with several data association hypotheses is to consider all possible hypotheses at each time step, and update the objects' state for each one of them. Hence, there will be a set of tracked objects with their corresponding state vector estimates for each possible sequence of data association hypotheses from the initial time step up to the current time step. Such a particular history of data association hypotheses is known as a global data association hypothesis. The advantages of this strategy is that all possibilities are considered, and as time progresses, the probabilities of the correct global data association hypothesis is expected to dominate, giving a good state estimate of the tracked objects.

In the case of the PMBM filter, the tracked objects $\mathcal{X}^t$ are modeled as a MBM with parameters

$$\{(w^j, \{(r^{j,i}, f^{j,i})\}_{i\in\mathbb{I}^j})\}_{i\in\mathbb{J}}\,, \tag{4.11}$$

where $\mathbb{J}$ is the index set for all possible global data association hypotheses, and for a given $j \in \mathbb{J}$, the corresponding MB component describes the probability distribution of the tracked targets for the global data association hypothesis, which are indexed by the set $\mathbb{I}^j$.

Moreover, the unknown objects $\mathcal{X}^u$ are modeled as a PPP with intensity $\lambda^u(\mathbf{x})$, which is independent of the RFS for the tracked objects $\mathcal{X}^t$. Therefore, the combined distribution of all objects $\mathcal{X}$, i.e., $\mathcal{X} = \mathcal{X}^u \uplus \mathcal{X}^t$, is said to be distributed as a "Poisson multi-Bernoulli mixture".

In particular, it follows from Proposition 3.6.1, Proposition 3.6.4 and Theorem 3.6.1 that the PGFL for the PMBM $\mathcal{X}$ is equal to

$$F_{\mathcal{X}}(h) = e^{-<D^u, h-1>} \sum_{j\in\mathbb{J}} w^j \prod_{i\in\mathbb{I}^j} \left(1 - r^{j,i} + r^{j,i} < f^{j,i}, h >\right). \tag{4.12}$$

Moreover, the PMBM density is univocally determined by the parameters

$$D^u, \{(w^j, \{(r^{j,i}, f^{j,i})\}_{i\in\mathbb{I}^j})\}_{i\in\mathbb{J}}. \tag{4.13}$$

4.4 PMBM prediction and update steps

Bayesian OT methods for a single object are based on two important equations: The Chapman-Kolmogorov equation, which is given by

$$f(\mathbf{x}_k|\mathbf{z}_{1:k-1}) = \int f_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1}) f(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})\, d\mathbf{x}_{k-1}\,, \tag{4.14}$$

and Bayes' rule, which is given by

$$f(\mathbf{x}_k|\mathbf{z}_{1:k}) = \frac{f(\mathbf{z}_k|\mathbf{x}_k)f(\mathbf{x}_k|\mathbf{z}_{1:k-1})}{\int f(\mathbf{z}_k|\mathbf{x}_k)f(\mathbf{x}_k|\mathbf{z}_{1:k-1})\,d\mathbf{x}_k} \tag{4.15a}$$

$$\propto f(\mathbf{z}_k|\mathbf{x}_k)f(\mathbf{x}_k|\mathbf{z}_{1:k-1})\,, \tag{4.15b}$$

The Chapman-Kolmogorov equation propagates the distribution of $\mathbf{x}$ at time step $k-1$ conditioned to the measurements up to time step $k-1$, $f(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})$, to the next time step using the state-transition density function $f_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$. This provides the distribution of $\mathbf{x}$ at time step k conditioned to the measurements up to time step $k-1$. Therefore, this equation gives the so-called "prediction" step of the target tracking method. After this step, the uncertainties in the estimate usually increase due to the addition of model uncertainties.

On the other hand, Bayes' rule provides the posterior distribution of $\mathbf{x}$ at time step k conditioned to the measurements up to that time step, $f(\mathbf{x}_{k-1}|\mathbf{z}_{1:k})$, by using the likelihood function $f(\mathbf{z}_k|\mathbf{x}_k)$ and the predicted distribution $f_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$. The step of the target tracking method based on Bayes' rule is known as the "update" step because it usually improves the state estimate since it uses the measurement model and the information of the additional measurement at time k.

In the case of the PMBM filter, the generalizations of the Chapman-Kolmogorov equation and Bayes' rule to RFSs are used

$$f(\mathcal{X}_k|\mathcal{Z}_{1:k-1}) = \int f_{k|k-1}(\mathcal{X}_k|\mathcal{X}_{k-1})f(\mathcal{X}_{k-1}|\mathcal{Z}_{1:k-1})\,\delta\mathcal{X}_{k-1} \tag{4.16a}$$

$$f(\mathcal{X}_k|\mathcal{Z}_{1:k}) \propto f(\mathcal{Z}_k|\mathcal{X}_k)f(\mathcal{X}_k|\mathcal{Z}_{1:k-1}).\,, \tag{4.16b}$$

where the concepts of set density function (Definition 3.2.1) and set integral (Definition 3.5.1) are central.

We will now prove in the following two theorems that the structure of the PMBM PGFL (4.12) is preserved by the application of the Chapman-Kolmogorov equation (4.16a and the Bayes' rule 4.16b) for RFS. In particular, by examining the parameter expressions of the obtained PGFLs, prediction and update laws are derived for the PMBM parameters (4.13).

Theorem 4.4.1. *(PMBM prediction step) Let the posterior distribution $\mathcal{X}_k$ be a PMBM with parameters*

$$D_k^u, \{w_k^j, \{(r_k^{j,i}, f_k^{j,i})\}_{i\in\mathbb{I}_k^j}\}_{j\in\mathbb{J}_k}. \tag{4.17}$$

*Then it follows from the dynamic model **(A.1)**-**(A.5)** and the Chapman-Kolmogorov equation for RFSs (4.16a) that the predicted distribution $\mathcal{X}_{k|k-1}$ is also a PMBM with parameters*

$$D_{k|k-1}^u, \{w_{k|k-1}^j, \{(r_{k|k-1}^{j,i}, f_{k|k-1}^{j,i})\}_{i\in\mathbb{I}_k^j}\}_{j\in\mathbb{J}_k}\,, \tag{4.18}$$

where

$$D^u_{k|k-1}(\mathbf{x}) = D^b(\mathbf{x}) + < D_k, p^s f_k(\mathbf{x}|\cdot) > \tag{4.19a}$$

$$w^j_{k|k-1} = w^j_k \tag{4.19b}$$

$$r^{j,i}_{k|k-1} = r^{j,i} < f^{j,i}_k, p^s > \tag{4.19c}$$

$$f^{j,i}_{k|k-1}(\mathbf{x}) = \frac{< f^{j,i}_k, p^s f_k(\mathbf{x}|\cdot) >}{< f^{j,i}_k, p^s >} \tag{4.19d}$$

and $f_k(\mathbf{x}|\cdot)$ is the function $f_k(\mathbf{x}|\mathbf{x}')$, where $\mathbf{x}$ is fixed and $\mathbf{x}'$ is variable.

In particular, the number of components in the MBM, as well as the number of Bernoulli components in each MB, remain unchanged by the PMBM prediction step.

Proof. The proof consists of the following steps:

1. Find the PGFL of the conditioned RFS $\mathcal{X}|\mathcal{X}'$, which corresponds to the transition from a RFS $\mathcal{X}'$ to the RFS $\mathcal{X}$ at the next time step.
2. Express the PGFL of the predicted distribution $\mathcal{X}_{k|k-1}$ as a function of the PGFL of the updated distribution $\mathcal{X}_k$ using the PGFL of $\mathcal{X}|\mathcal{X}'$ found in the previous step.
3. Insert the PMBM expression for the PGFL of the updated distribution $\mathcal{X}_k$ into the formula found in the previous step. Then, verify that the resulting expression corresponds to the PGFL of a PMBM. By doing so, the prediction laws for the parameters (4.19) are also derived.

Indeed,

1. The transition RFS $\mathcal{X}|\mathcal{X}'$ consists of two independent processes: The birth of new objects and the survival and state-transition of existing objects (**A.1**)-(**A.5**).

   According to (**A.2**), the birth of new objects is a PPP with birth intensity $D^b(\mathbf{x})$. Hence, it follows from Proposition 3.6.1, that its PGFL is given by

   $$F^b_{\mathcal{X}|\mathcal{X}'}(h|X') = e^{<D^b, h-1>}. \tag{4.20}$$

   where X' is a realization of $\mathcal{X}'$. The need to specify the dependence of $F^b_{\mathcal{X}|\mathcal{X}'}$ on X' is necessary since it is part of the PGFL of a conditioned RFS. However, in this case, its actual expression is independent of any realization.

   On the hand, due to (**A.3**) and (**A.4**), the survival of an existing object and the state-transition of this object in the case that it survives, constitutes a Bernoulli process with existence probability $p^s(\mathbf{x}')$ and spatial distribution $f_{k|k-1}(\cdot|\mathbf{x}')$, where $\mathbf{x}'$ is its state vector before the transition and $f_k(\cdot|\mathbf{x}')$ is the function $f_k(\mathbf{x}|\mathbf{x}')$, where $\mathbf{x}$ is variable and $\mathbf{x}'$ is fixed.

   Furthermore, (**A.3**) and (**A.4**) ensure that these Bernoulli processes are independent. Therefore, the survival and transition for all existing objects constitutes a MB process. Hence, it follows from Proposition 3.6.3, that its corresponding PGFL is given by

   $$F^{s,t}_{\mathcal{X}|\mathcal{X}'}(h|X') = \prod_{\mathbf{x}'\in X'} \left(1 - p^s(\mathbf{x}') + p^s(\mathbf{x}') < f_{k|k-1}(\cdot|\mathbf{x}'), h > \right), \tag{4.21}$$

where X' is a realization of $\mathcal{X}'$.

Finally, since the birth of new targets and the process of survival and state-transition are independent processes according to (**A.5**), Theorem 3.6.1 yields the PGFL of

$$F_{\mathcal{X}|\mathcal{X}'}(h|X') = F^{b}_{\mathcal{X}|\mathcal{X}'}(h|X')F^{s,t}_{\mathcal{X}|\mathcal{X}'}(h|X') \tag{4.22a}$$

$$= e^{<D^b,h-1>} \prod_{\mathbf{x}'\in X'} (1 - p^s(\mathbf{x}') + p^s(\mathbf{x}') < f_{k|k-1}(\cdot|\mathbf{x}'), h >). \tag{4.22b}$$

2. The PGFL of the predicted distribution $\mathcal{X}_{k|k-1}$ is by definition

$$F_{k|k-1}(h) = \int h^X f_{k|k-1}(X)\, \delta X\,, \tag{4.23}$$

where $f_{k|k-1}$ is its corresponding set density function.

By inserting Chapman-Kolmogorovs equation for RFSs (4.16a) into (4.23) and rearranging the integrand and the order of integration, we obtain that

$$F_{k|k-1}(h) = \int h^X f_{k|k-1}(X)\, \delta X \tag{4.24a}$$

$$= \int h^X \int f_{\mathcal{X}|\mathcal{X}'}(X|X') f_k(X')\, \delta X'\, \delta X \tag{4.24b}$$

$$= \int \int h^X f_{\mathcal{X}|\mathcal{X}'}(X|X')\, \delta X\, f_k(X')\, \delta X' \tag{4.24c}$$

$$= \int F_{\mathcal{X}|\mathcal{X}'}(h|X') f_k(X')\, \delta X'. \tag{4.24d}$$

Finally, by replacing the expression for $F_{\mathcal{X}|\mathcal{X}'}$ (4.22) into (4.24), and using that

$$\mathbf{x}' \to 1 - p^s(\mathbf{x}') + p^s(\mathbf{x}') < f_{k|k-1}(\cdot|\mathbf{x}'), h >) \tag{4.25}$$

is a test function, the expression that relates the PGFLs $F_{k|k-1}$ and F_k follows from the definition of a PGFL:

$$F_{k|k-1}(h) = \int F_{\mathcal{X}|\mathcal{X}'}(h|X') f_k(X')\, \delta X' \tag{4.26a}$$

$$= \int e^{<D^b,h-1>} \prod_{\mathbf{x}'\in X'} (1 - p^s(\mathbf{x}') + p^s(\mathbf{x}') < f_{k|k-1}(\cdot|\mathbf{x}'), h >) f_k(X')\, \delta X' \tag{4.26b}$$

$$= e^{<D^b,h-1>} \int \prod_{\mathbf{x}'\in X'} (1 - p^s(\mathbf{x}') + p^s(\mathbf{x}') < f_{k|k-1}(\cdot|\mathbf{x}'), h >) f_k(X')\, \delta X' \tag{4.26c}$$

$$= e^{<D^b,h-1>} \int (1 - p^s(\mathbf{x}') + p^s(\mathbf{x}') < f_{k|k-1}(\cdot|\mathbf{x}'), h >)^{X'} f_k(X')\, \delta X' \tag{4.26d}$$

$$= e^{<D^b,h-1>} F_k(1 - p^s(\mathbf{x}') + p^s(\mathbf{x}') < f_{k|k-1}(\cdot|\mathbf{x}'), h >). \tag{4.26e}$$

3. By hypothesis the updated distribution $\mathcal{X}_k$ is a PMBM with parameters

$$D_k^u, \{w_k^j, \{(r_k^{j,i}, f_k^{j,i})\}_{i\in\mathbb{I}_k^j}\}_{j\in\mathbb{J}_k}. \tag{4.27}$$

Hence, by Proposition 3.6.1, Proposition 3.6.4 and Theorem 3.6.1, the PGFL of $\mathcal{X}_k$ is given by

$$F_{\mathcal{X}_k} = F_{\mathcal{X}_k}^{\text{PPP}} F_{\mathcal{X}_k}^{\text{MBM}} \tag{4.28}$$

where

$$F_{\mathcal{X}_k}^{\text{PPP}}(h) = e^{<D_k, h-1>} \tag{4.29a}$$

$$F_{\mathcal{X}_k}^{\text{MBM}}(h) = \sum_{j\in\mathbb{J}_k} w_k^j \prod_{i\in\mathbb{I}_k^j} \left(1 - r_k^{j,i} + r_k^{j,i} < f_k^{j,i}, h >\right). \tag{4.29b}$$

By inserting (4.28) into (4.26), we obtain that

$$\begin{aligned} F_{k|k-1}(h) = e^{<D^b, h-1>} F_{\mathcal{X}_k}^{\text{PPP}}(1 - p^s(\mathbf{x}') + p^s(\mathbf{x}') < f_{k|k-1}(\cdot|\mathbf{x}'), h >) \dots \\ \cdots \times F_{\mathcal{X}_k}^{\text{MBM}}(1 - p^s(\mathbf{x}') + p^s(\mathbf{x}') < f_{k|k-1}(\cdot|\mathbf{x}'), h >). \end{aligned} \tag{4.30a}$$

We will now show that the product on the right-hand side of (4.30a) corresponds to the PGFL of a PMBM. More precisely, the product of the two first terms is the PGFL of a PPP, while the remaining term is the PGFL of a MBM. Moreover, while showing this, expressions for the PMBM parameters will be derived.

The product of the two first terms on the right-hand side of (4.30a) is

$$e^{<D^b, h-1>} F_{\mathcal{X}_k}^{\text{PPP}}(1 - p^s(\mathbf{x}') + p^s(\mathbf{x}') < f_{k|k-1}(\cdot|\mathbf{x}'), h >) \dots \tag{4.31a}$$

$$= e^{<D^b, h-1>} e^{<D_k^u, 1 - p^s(\mathbf{x}') + p^s(\mathbf{x}') < f_{k|k-1}(\cdot|\mathbf{x}'), h > - 1>}. \tag{4.31b}$$

By using that the integral of $f_{k|k-1}(\mathbf{x}|\mathbf{x}')$ with respect to $\mathbf{x}$ and rearranging the integrand and the order of integration, the exponent of the second factor in (4.31b) can be rewritten as

$$< D_k^u, p^s(\mathbf{x}') + p^s(\mathbf{x}') < f_{k|k-1}(\cdot|\mathbf{x}'), h >> \tag{4.32a}$$

$$= \int D_k^u(\mathbf{x}') \left(-p^s(\mathbf{x}') + p^s(\mathbf{x}') \int f_{k|k-1}(\mathbf{x}|\mathbf{x}') h(\mathbf{x})\, d\mathbf{x}\right) d\mathbf{x}' \tag{4.32b}$$

$$= \int D_k^u(\mathbf{x}') \left(-p^s(\mathbf{x}') \int f_{k|k-1}(\mathbf{x}|\mathbf{x}')\, d\mathbf{x} + p^s(\mathbf{x}') \int f_{k|k-1}(\mathbf{x}|\mathbf{x}') h(\mathbf{x})\, d\mathbf{x}\right) d\mathbf{x}' \tag{4.32c}$$

$$= \int \int D_k^u(\mathbf{x}') p^s(\mathbf{x}') f_{k|k-1}(\mathbf{x}|\mathbf{x}') \left(h(\mathbf{x}) - 1\right) d\mathbf{x}\, d\mathbf{x}' \tag{4.32d}$$

$$= \int \left(\int D_k^u(\mathbf{x}') p^s(\mathbf{x}') f_{k|k-1}(\mathbf{x}|\mathbf{x}')\right) d\mathbf{x}' \left(h(\mathbf{x}) - 1\right) d\mathbf{x} \tag{4.32e}$$

$$=<< D_k^u, p^s f_{k|k-1}(\mathbf{x}|\cdot) >, h - 1 > . \tag{4.32f}$$

By inserting (4.32f) into (eq. (4.31)), the product of the two first terms on the right-hand side of (4.30a) becomes

$$e^{<D^b,h-1>}F^{\text{PPP}}_{\mathcal{X}_k}(1-p^s(\mathbf{x}')+p^s(\mathbf{x}')<f_{k|k-1}(\cdot|\mathbf{x}'),h>)\dots \tag{4.33a}$$

$$=e^{<D^b,h-1>}e^{<<D^u_k,p^s f_{k|k-1}(\mathbf{x}|\cdot)>,h-1>} \tag{4.33b}$$

$$=e^{<D^b+<D^u_k,p^s f_{k|k-1}(\mathbf{x}|\cdot)>,h-1>}\,, \tag{4.33c}$$

and it is immediate to verify that (4.33c) is the PGFL of a PPP (3.44) with intensity

$$D^u_{k|k-1}(\mathbf{x})=D^b(\mathbf{x})+<D_k,p^s f_k(\mathbf{x}|\cdot)>\,. \tag{4.34}$$

Consider now the third factor on the right-hand side of (4.30a). By (4.29b), it is equal to

$$F^{\text{MBM}}_{\mathcal{X}_k}(1-p^s(\mathbf{x}')+p^s(\mathbf{x}')<f_{k|k-1}(\cdot|\mathbf{x}'),h>) \tag{4.35a}$$

$$=\sum_{j\in\mathbb{J}_k}w^j_k\prod_{i\in\mathbb{I}^j_k}\left(1-r^{j,i}_k+r^{j,i}_k<f^{j,i}_k,1-p^s(\mathbf{x}')+p^s(\mathbf{x}')<f_{k|k-1}(\cdot|\mathbf{x}'),h>>\right). \tag{4.35b}$$

By using that the integral of $f_{k|k-1}(\mathbf{x}|\mathbf{x}')$ with respect to $\mathbf{x}$ and rearranging the integrand and the order of integration, the factors in (4.35b) can be rewritten as

$$1-r^{j,i}_k+r^{j,i}_k<f^{j,i}_k,1-p^s(\mathbf{x}')+p^s(\mathbf{x}')<f_{k|k-1}(\cdot|\mathbf{x}'),h>> \tag{4.36a}$$

$$=1-r^{j,i}_k+r^{j,i}_k\int f^{j,i}_k(\mathbf{x}')\left(1-p^s(\mathbf{x}')+p^s(\mathbf{x}')\int f_{k|k-1}(\mathbf{x}|\mathbf{x}')h(\mathbf{x})\,d\mathbf{x}\right)d\mathbf{x}' \tag{4.36b}$$

$$=1-r^{j,i}_k+r^{j,i}_k\int f^{j,i}_k(\mathbf{x}')\,d\mathbf{x}'-\int f^{j,i}_k(\mathbf{x}')p^s(\mathbf{x}')\,d\mathbf{x}'\dots$$

$$\cdots+\int f^{j,i}_k(\mathbf{x}')p^s(\mathbf{x}')\int f_{k|k-1}(\mathbf{x}|\mathbf{x}')h(\mathbf{x})\,d\mathbf{x}\,d\mathbf{x}' \tag{4.36c}$$

$$=1-r^{j,i}_k+r^{j,i}_k-<f^{j,i}_k,p^s>+\int\left(\int f^{j,i}_k(\mathbf{x}')p^s(\mathbf{x}')f_{k|k-1}(\mathbf{x}|\mathbf{x}')\,d\mathbf{x}'\right)h(\mathbf{x})d\mathbf{x} \tag{4.36d}$$

$$=1-<f^{j,i}_k,p^s>+<<f^{j,i}_k,p^s f_{k|k-1}(\mathbf{x}|\cdot)>,h> \tag{4.36e}$$

$$=1-<f^{j,i}_k,p^s>+<f^{j,i}_k,p^s>\left\langle\frac{<f^{j,i}_k,p^s f_{k|k-1}(\mathbf{x}|\cdot)>}{<f^{j,i}_k,p^s>},h\right\rangle. \tag{4.36f}$$

Hence, the expression in (4.35b) for the third factor on the right-hand side of (4.30a) (4.35b) it is equal to

$$F^{\text{MBM}}_{\mathcal{X}_k}(1-p^s(\mathbf{x}')+p^s(\mathbf{x}')<f_{k|k-1}(\mathbf{x}|\cdot),h>) \tag{4.37a}$$

$$=\sum_{j\in\mathbb{J}_k}w^j_k\prod_{i\in\mathbb{I}^j_k}\left(1-<f^{j,i}_k,p^s>+<f^{j,i}_k,p^s>\left\langle\frac{<f^{j,i}_k,p^s f_{k|k-1}(\mathbf{x}|\cdot)>}{<f^{j,i}_k,p^s>},h\right\rangle\right), \tag{4.37b}$$

and it is immediate to verify that (4.37b) is the PGFL of a MBM (3.51) with

$$\mathbb{J}_{k|k-1} = \mathbb{J}_k \tag{4.38a}$$

$$\mathbb{I}^j_{k|k-1} = \mathbb{I}^j_k \tag{4.38b}$$

$$w^j_{k|k-1} = w^j_k \tag{4.38c}$$

$$r^{j,i}_{k|k-1} = r^{j,i}_k < f^{j,i}_k, p^s > \tag{4.38d}$$

$$f^{j,i}_{k|k-1}(\mathbf{x}) = \frac{< f^{j,i}_k, p^s f_k(\mathbf{x}|\cdot) >}{< f^{j,i}_k, p^s >}. \tag{4.38e}$$

□

Theorem 4.4.2. *(PMBM update step) Let the predicted distribution $\mathcal{X}_{k|k-1}$ be a PMBM with parameters*

$$D^u_{k|k-1}, \{w^j_{k|k-1}, \{(r^{j,i}_{k|k-1}, f^{j,i}_{k|k-1})\}_{i\in\mathbb{I}^j_{k|k-1}}\}_{j\in\mathbb{J}_{k|k-1}}, \tag{4.39}$$

*and let Z_k be the set of measurements obtained at the time step k. Then it follows from the dynamic model (**A.1**)-(**A.5**), the measurement model (**B.1**)-(**B.6**) and the Bayes rule for RFSs (4.16b) that the posterior distribution $\mathcal{X}_{k|k-1}|\mathcal{Z}_k$ is a PMBM with parameters*

$$D^u_k, \{w^{j,A}_k, \{(r^{j,C}_k, f^{j,C}_k)\}_{C\in A}\}_{j\in\mathbb{J}_{k|k-1}, A\in\mathcal{A}^j}, \tag{4.40}$$

where $\mathcal{A}^j$ is the set of all possible data associations for the global hypothesis with index j, the updated intensity of the PPP for the unknown objects is

$$D^u_k(\mathbf{x}) = q^d(\mathbf{x}) D^u_{k|k-1}(\mathbf{x}), \tag{4.41}$$

the weights of the MBM for the tracked objects are

$$w^{j,A}_k = \frac{w^j_{k|k-1}\prod_{C\in A} L^{j,C}}{\sum_{j\in\mathbb{J}_{k|k-1}}\sum_{A\in\mathcal{A}^j} w^j_{k|k-1}\prod_{C\in A} L^{j,C}}, \tag{4.42a}$$

$$L^{j,C} = \begin{cases} D^c(\mathbf{z}_c) + < D^u_{k|k-1}, l_C > & \text{, if } C\cap\mathbb{I}^j = \emptyset,\ Z_C = \{\mathbf{z}_c\} \neq \emptyset \\ < D^u_{k|k-1}, l_C > & \text{, if } C\cap\mathbb{I}^j = \emptyset,\ |Z_C| > 1 \\ 1 - r^{j,i_C}_{k|k-1} + r^{j,i_C}_{k|k-1} < f^{j,i_C}_{k|k-1}, q^d > & \text{, if } C\cap\mathbb{I}^j = \{i_C\},\ Z_C = \emptyset \\ r^{j,i_C}_{k|k-1} < f^{j,i_C}_{k|k-1}, q^d > & \text{, if } C\cap\mathbb{I}^j = \{i_C\},\ Z_C \neq \emptyset, \end{cases} \tag{4.42b}$$

and the parameters of the Bernoulli components are

$$r^{j,C}_k = \begin{cases} \frac{<D^u_{k|k-1}, l_C>}{D^c(\mathbf{z}_c) + <D^u_{k|k-1}, l_C>} & \text{, if } C\cap\mathbb{I}^j = \emptyset,\ Z_C = \{\mathbf{z}_c\} \neq \emptyset \\ 1 & \text{, if } C\cap\mathbb{I}^j = \emptyset,\ |Z_C| > 1 \\ \frac{r^{j,i_C}_{k|k-1} < f^{j,i_C}_{k|k-1}, q^d>}{1 - r^{j,i_C}_{k|k-1} + r^{j,i_C}_{k|k-1} < f^{j,i_C}_{k|k-1}, q^d>} & \text{, if } C\cap\mathbb{I}^j = \{i_C\},\ Z_C = \emptyset \\ 1 & \text{, if } C\cap\mathbb{I}^j = \{i_C\},\ Z_C \neq \emptyset, \end{cases} \tag{4.43a}$$

$$f^{j,C}_k(\mathbf{x}) = \begin{cases} \frac{l_C(\mathbf{x}) D^u_{k|k-1}(\mathbf{x})}{<l_C, D^u_{k|k-1}>} & \text{, if } C\cap\mathbb{I}^j = \emptyset \\ \frac{q^d(\mathbf{x}) f^{j,i_C}_{k|k-1}(\mathbf{x})}{<q^d, f^{j,i_C}_{k|k-1}>} & \text{, if } C\cap\mathbb{I}^j = \{i_C\},\ Z_C = \emptyset \\ \frac{l_C(\mathbf{x}) f^{j,i_C}_{k|k-1}(\mathbf{x})}{<l_C, f^{j,i_C}_{k|k-1}>} & \text{, if } C\cap\mathbb{I}^j = \{i_C\},\ Z_C \neq \emptyset. \end{cases} \tag{4.43b}$$

In particular, the number of components in the updated MBM increases with one MB process for each MB process in the predicted MBM ($j \in \mathbb{J}$) and for each possible data association given the measurements Z_k ($A \in \mathcal{A}^j$). Moreover, each MB component consists of as many Bernoulli processes as cells there are cells in its corresponding data association.

The prove of this theorem is complex. Therefore, we prove some useful lemmas first.

Lemma 4.4.1. *Let F(g,h) be the functional defined as*

$$F(g,h) =< D^c, g-1 > + < D^u_{k|k-1}, h\left(1 - p^d + p^d e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}), g-1>}\right) - 1 >, \tag{4.44}$$

where $g(\mathbf{z})$ and $h(\mathbf{x})$ are test functions of variable $\mathbf{z}$ and $\mathbf{x}$, respectively.

Then the functional derivative of F with respect to the vector $\mathbf{z}$ is

$$\frac{\delta F(g,h)}{\delta \mathbf{z}} = \left(D^c(\mathbf{z}) + \left\langle D^u_{k|k-1}, h l_{\{\mathbf{z}\}} e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}), g>}\right\rangle\right). \tag{4.45}$$

Furthermore, the functional derivative of (4.45) respect to a non-empty set Z that does not contain $\mathbf{z}$ is

$$\frac{\delta\left(D^c(\mathbf{z}) + \left\langle D^u_{k|k-1}, h l_{\{\mathbf{z}\}} e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}), g>}\right\rangle\right)}{\delta Z} = \left\langle D^u_{k|k-1}, h l_{\{\mathbf{z}\}\cup Z} e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}), g>}\right\rangle. \tag{4.46}$$

Proof. By using the linearity of the functional derivative and the bilinearity of the scalar product of functions, the functional F can be written as

$$\begin{aligned} F(g,h) =& < D^c, g-1 > + < D^u_{k|k-1}, h\left(1 - p^d + p^d e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}), g-1>}\right) - 1 > && \text{(4.47a)} \\ =& < D^c, g > + < D^u_{k|k-1}, h p^d e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}), g-1>} > \ldots && \text{(4.47b)} \\ & \cdots - < D^c, 1 > + < D^u_{k|k-1}, h\left(1 - p^d +\right) - 1 >, && \text{(4.47c)} \end{aligned}$$

where the terms in (4.47c) do not depend on the test function $g(\mathbf{z})$. Therefore, the functional derivative of these terms is zero, and the functional derivative of F simplifies to

$$\frac{\delta F(g,h)}{\delta \mathbf{z}} = \left(\frac{\delta < D^c, g >}{\delta \mathbf{z}} + \frac{\delta < D^u_{k|k-1}, h p^d e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}), g>} >}{\delta \mathbf{z}}\right). \tag{4.48}$$

The functional derivative of the first term in the parenthesis of (4.48) follows from the definition of functional derivative (3.6.3) and the linearity of the scalar product of functions, which gives

$$\begin{aligned} \frac{\delta < D^c, g >}{\delta \mathbf{z}} &= \lim_{\epsilon \to 0^+} \frac{< D^c, g + \epsilon\delta_{\mathbf{z}} > - < D^c, g >}{\epsilon} && \text{(4.49a)} \\ &= \lim_{\epsilon \to 0^+} \frac{< D^c, \epsilon\delta_{\mathbf{z}} >}{\epsilon} && \text{(4.49b)} \\ &=< D^c, \delta_{\mathbf{z}} >= D^c(\mathbf{z}). && \text{(4.49c)} \end{aligned}$$

The functional derivative of the second term in the parenthesis of (4.48) is more involved, and it requires also the use of continuity of the scalar product of functions and the continuity of

the exponential function. By using these properties, we obtain that

$$\frac{\delta < D^u_{k|k-1}, hp^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} >}{\delta \mathbf{z}} \tag{4.50a}$$

$$= \lim_{\epsilon \to 0^+} \frac{< D^u_{k|k-1}, hp^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g+\epsilon\delta_{\mathbf{z}}>} > - < D^u_{k|k-1}, hp^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} >}{\epsilon} \tag{4.50b}$$

$$= \left\langle D^u_{k|k-1}, hp^d \lim_{\epsilon \to 0^+} \frac{e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g+\epsilon\delta_{\mathbf{z}}>} - e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>}}{\epsilon} \right\rangle \tag{4.50c}$$

$$= \left\langle D^u_{k|k-1}, hp^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} \lim_{\epsilon \to 0^+} \frac{< \lambda^m(\mathbf{x})l(\cdot|\mathbf{x}), g + \epsilon\delta_{\mathbf{z}} > - < \lambda^m(\mathbf{x})l(\cdot|\mathbf{x}), g >}{\epsilon} \right\rangle \tag{4.50d}$$

$$= \left\langle D^u_{k|k-1}, hp^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} \lim_{\epsilon \to 0^+} \frac{< \lambda^m(\mathbf{x})l(\cdot|\mathbf{x}), \epsilon\delta_{\mathbf{z}} >}{\epsilon} \right\rangle \tag{4.50e}$$

$$= \left\langle D^u_{k|k-1}, hp^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} < \lambda^m(\mathbf{x})l(\cdot|\mathbf{x}), \delta_{\mathbf{z}} > \right\rangle \tag{4.50f}$$

$$= \left\langle D^u_{k|k-1}, hp^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} \lambda^m(\mathbf{x})l(\mathbf{z}|\mathbf{x}) \right\rangle \tag{4.50g}$$

$$= \left\langle D^u_{k|k-1}, hl_{\{\mathbf{z}\}} e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} \right\rangle , \tag{4.50h}$$

where the last equality follows from the definition of l_Z (4.3).

We proceed now to prove the second part of the lemma.

Let $\mathbf{z}'$ be a vector measurement different from $\mathbf{z}$. The linearity of the functional derivative and the fact that $D^c(\mathbf{z})$ does not depend on g yields that

$$\frac{\delta \left(D^c(\mathbf{z}) + \left\langle D^u_{k|k-1}, hl_{\{\mathbf{z}\}} e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} \right\rangle \right)}{\delta \mathbf{z}'} = \frac{\delta \left\langle D^u_{k|k-1}, hl_{\{\mathbf{z}\}} e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} \right\rangle}{\delta \mathbf{z}'}. \tag{4.51}$$

Now, we observe that the calculation of the functional derivative in (4.51) is analogous to the calculations performed in (4.50) due to the similar structure of the functionals. By following the steps in (4.50), we conclude that

$$\frac{\delta \left(D^c(\mathbf{z}) + \left\langle D^u_{k|k-1}, hl_{\{\mathbf{z}\}} e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} \right\rangle \right)}{\delta \mathbf{z}'} = \frac{\delta \left\langle D^u_{k|k-1}, hl_{\{\mathbf{z}\}} e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} \right\rangle}{\delta \mathbf{z}'} \tag{4.52a}$$

$$= \left\langle D^u_{k|k-1}, hl_{\{\mathbf{z}\}} \lambda^m(\mathbf{x})l(\mathbf{z}'|\mathbf{x}) e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} \right\rangle \tag{4.52b}$$

$$= \left\langle D^u_{k|k-1}, hl_{\{\mathbf{z},\mathbf{z}'\}} e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} \right\rangle \tag{4.52c}$$

In particular, we observe that consecutive functional derivations of (eq. (4.52)) with respect to other vector measurements, only increments the number of elements in the likelihood term. In other words,

$$\frac{\delta^2 \left(D^c(\mathbf{z}) + \left\langle D^u_{k|k-1}, hl_{\{\mathbf{z}\}} e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} \right\rangle \right)}{\delta \mathbf{z}'' \delta \mathbf{z}'} = \left\langle D^u_{k|k-1}, hl_{\{\mathbf{z},\mathbf{z}',\mathbf{z}''\}} e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} \right\rangle \tag{4.53a}$$

$$\frac{\delta^3 \left(D^c(\mathbf{z}) + \left\langle D^u_{k|k-1}, hl_{\{\mathbf{z}\}} e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} \right\rangle \right)}{\delta \mathbf{z}''' \delta \mathbf{z}'' \delta \mathbf{z}'} = \left\langle D^u_{k|k-1}, hl_{\{\mathbf{z},\mathbf{z}',\mathbf{z}'',\mathbf{z}'''\}} e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>} \right\rangle \tag{4.53b}$$

and so on. Therefore, we conclude by induction that

$$\frac{\delta\left(D^c(\mathbf{z})+\left\langle D^u_{k|k-1}, hl_{\{\mathbf{z}\}}e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>}\right\rangle\right)}{\delta Z}=\left\langle D^u_{k|k-1}, hl_{\{\mathbf{z}\}\cup Z}e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>}\right\rangle \tag{4.54}$$

for any non-empty set Z that does not contain the original measurement vector $\mathbf{z}$. □

Lemma 4.4.2. *Let $F^{j,i}(g,h)$ be the functional defined as*

$$F^{j,i}(g,h)=1-r^{j,i}_{k|k-1}+r^{j,i}_{k|k-1}<f^{j,i}_{k|k-1},h\left(1-p^d+p^de^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g-1>}\right)>, \tag{4.55}$$

where $g(\mathbf{z})$ and $h(\mathbf{x})$ are test functions of variable $\mathbf{z}$ and $\mathbf{x}$, respectively.

Then the functional derivative of $F^{j,i}(g,h)$ with respect to a non-empty set Z is

$$\frac{\delta F^{j,i}}{\delta Z}(g,h)=r^{j,i}_{k|k-1}\left\langle h, f^{j,i}l_Ze^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g>}\right\rangle. \tag{4.56}$$

Proof. The only term of $F^{j,i}$ (4.55)that depends on g:

$$r^{j,i}_{k|k-1}<f^{j,i}_{k|k-1},h\left(1-p^d+p^de^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g-1>}\right)>, \tag{4.57}$$

has similar structure to the functional that is derived in the second part of Lemma 4.4.1. Therefore, the proof of this lemma is analogous to the one for Lemma 4.4.1, and is omited to avoid repetition. □

Lemma 4.4.3. *Any functional of the form $F(h)=a+<b,h>$, where $a\geq 0$ is a non-negative scalar and $b(\mathbf{x})\geq 0$ is a non-negative integrable function with $<b,1>=\int b(\mathbf{x})\,d\mathbf{x}>0$, can be written as*

$$F(h)=a+<b,h>=L\left(1-r+r<f,h>\right), \tag{4.58}$$

where

$$L=a+<b,1> \tag{4.59a}$$

$$r=\frac{<b,1>}{a+<b,1>} \tag{4.59b}$$

$$f(\mathbf{x})=\frac{b(\mathbf{x})}{<b,1>}. \tag{4.59c}$$

In particular, the function F corresponds to an unnormalized Bernoulli distribution.

Proof. Since $a\geq 0$ and $<b,1>0$, the scalar r and the function $f(\mathbf{x})$ are well-defined.

The result follows by inserting the expressions in (4.59) into (4.58):

$$L\left(1-r+r<f,h>\right) \tag{4.60a}$$

$$=(a+<b,1>)\left(1-\frac{<b,1>}{a+<b,1>}+\frac{<b,1>}{a+<b,1>}\left\langle\frac{b}{<b,1>},h\right\rangle\right) \tag{4.60b}$$

$$=a+<b,1>-<b,1>+<b,h>=a+<b,h>=F(h). \tag{4.60c}$$

□

Proof of Theorem 4.4.2. The proof consists of the following steps:

1. Express the PGFL of the updated distribution $\mathcal{X}_k|\mathcal{Z}_k$ as the functional derivative of the PGFL of the joint distribution $(\mathcal{Z}_k, \mathcal{X}_k)$.
2. Find the PGFL of the likelihood $\mathcal{Z}_k|\mathcal{X}_k$.
3. Use the expression found in the previous step, to write the PGFL of the joint distribution $(\mathcal{Z}_k, \mathcal{X}_k)$ as a function of the PGFL of the predicted distribution $\mathcal{X}_{k|k-1}$.
4. Insert the PMBM expression for the PGFL of the predicted distribution $\mathcal{X}_{k|k-1}$ into the formula found in the previous step, and divide the resulting expression into terms.
5. Use the linearity and product rule for PGFLs (3.6.3) to find the functional derivative of the terms defined in the previous step.
6. Find the PFGL of the updated distribution $\mathcal{X}_k|\mathcal{Z}_k$ by using the functional derivatives found in the previous step, and identify the resulting expression as a PMBM with the stated parameter values.

Indeed,

1. The PGFL of the updated distribution $\mathcal{X}_k|\mathcal{Z}_k$ is by definition

$$F_{\mathcal{X}_k|\mathcal{Z}_k}(h|Z_k) = \int h^X f_{\mathcal{X}_k|\mathcal{Z}_k}(X)\,\delta X\,, \tag{4.61}$$

where $f_{\mathcal{X}_k|\mathcal{Z}_k}$ is its corresponding set density function.

By applying Bayes' rule for RFSs (4.16b), we obtain that the PGFl for $\mathcal{X}_k$ is proportional to

$$F_{\mathcal{X}_k}(h|Z_k) = \int h^X f_{\mathcal{X}_k|\mathcal{Z}_k}(X|Z_k)\,\delta X \tag{4.62a}$$

$$\propto \int h^X f_{\mathcal{Z}_k|\mathcal{X}_k}(Z_k|X) f_{k|k-1}(X)\,\delta X\,, \tag{4.62b}$$

where $f_{\mathcal{Z}_k|\mathcal{X}_k}$ is the set density function of the likelihood $\mathcal{Z}_k|\mathcal{X}_k$.

Moreover, by Bayes' rule for RFSs (4.16b), the set function $Z_k \to f_{\mathcal{Z}_k|\mathcal{X}_k}(Z_k|X) f_{k|k-1}(X)$ is proportional to the set density function $f_{\mathcal{Z}_k|\mathcal{X}_k}$. Hence, it follows from Theorem 3.6.2 that

$$f_{\mathcal{Z}_k|\mathcal{X}_k}(Z_k|X) f_{k|k-1}(X) = \frac{\delta}{\delta Z_k} \int g^Z f_{\mathcal{Z}_k|\mathcal{X}_k}(Z|X) f_{k|k-1}(X)\,\delta \mathbf{z}\Big|_{g=0}\,, \tag{4.63}$$

where g is a test function.

By inserting (4.63) into (4.62b) and rearranging the order of integration and derivation, the PGFL of the updated distribution can be rewritten as

$$F_{\mathcal{X}_k}(h|Z_k) \propto \int h^X f_{\mathcal{Z}_k|\mathcal{X}_k}(Z_k|X) f_{k|k-1}(X)\,\delta X \tag{4.64a}$$

$$= \int h^X \frac{\delta}{\delta \mathbf{z}_k} \int g^Z f_{\mathcal{Z}_k|\mathcal{X}_k}(Z|X) f_{k|k-1}(X)\,\delta \mathbf{z}\Big|_{g=0} \delta X \tag{4.64b}$$

$$= \frac{\delta}{\delta Z_k} \int \int g^Z h^X f_{\mathcal{Z}_k|\mathcal{X}_k}(Z|X) f_{k|k-1}(X)\,\delta X\,\delta \mathbf{z}\Big|_{g=0}\,., \tag{4.64c}$$

where the integral in (4.64c) is defined as the PGFL for the joint distribution $(\mathcal{Z}_k, \mathcal{X}_k)$, $F_{(\mathcal{Z}_k,\mathcal{X}_k)}$. Hence,

$$F_{\mathcal{X}_k}(h|Z_k) \propto \frac{\delta}{\delta Z_k} F_{(\mathcal{Z}_k,\mathcal{X}_k)}(g,h)\Big|_{g=0}. \tag{4.65}$$

with

$$F_{(\mathcal{Z}_k,\mathcal{X}_k)}(g,h) = \int\int g^Z h^X f_{\mathcal{Z}_k|\mathcal{X}_k}(Z|X) f_{k|k-1}(X)\, \delta X\, \delta \mathbf{z}. \tag{4.66}$$

2. The likelihood RFS $\mathcal{Z}_k|\mathcal{X}_k$ depends on two independent processes: The generation of clutter and the generation of measurements from objects (**B.1**)-(**B.6**).

   According to (**B.2**), clutter is generated according to a PPP with birth intensity $D^c(\mathbf{x})$. Hence, it follows from Proposition 3.6.1, that its PGFL is given by

   $$F^c_{\mathcal{Z}_k|\mathcal{X}_k}(g|X) = e^{<D^c, g-1>}. \tag{4.67}$$

   On the hand, due to (**B.3**) and (**B.4**), the detection of an existing object with state vector $\mathbf{x}$ and the generation of measurements in the case that the object is detected, can be seen as a Bernoulli process with existence probability $p^d(\mathbf{x})$ and with a spatial distribution giving by a PPP with intensity $\lambda^m(\mathbf{x})l(\mathbf{z}|\mathbf{x})$. Therefore, as argumented in [14], the PGFL for this pseudo Bernoulli process is given by

   $$g \to 1 - p^d(\mathbf{x}) + p^d(\mathbf{x}) e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}), g-1>}, \tag{4.68}$$

   which combines elements of the PGFL of a Bernoulli process (3.47) and of the PGFL of a PPP (3.44).

   Since (**B.2**) ensures that these pseudo Bernoulli processes are independent, we obtain the PGFL for the the detection and measurement generating process for all existing objects by applying Theorem 3.6.1. This yields

   $$F^{d,g}_{\mathcal{Z}_k|\mathcal{X}_k}(g|X) = \prod_{\mathbf{x}\in X} \left(1 - p^d(\mathbf{x}) + p^d(\mathbf{x}) e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}), g-1>}\right). \tag{4.69}$$

   Finally, since the clutter generation and measurement generation are independent processes according to (**B.2**), by applying again Theorem 3.6.1, we obtain the PGFL for the likelihood:

   $$\begin{aligned} F_{\mathcal{Z}_k|\mathcal{X}_k}(g|X) &= F^c_{\mathcal{Z}_k|\mathcal{X}_k}(g|X) F^{d,g}_{\mathcal{Z}_k|\mathcal{X}_k}(g|X) && \text{(4.70a)} \\ &= e^{<D^c, g-1>} \prod_{\mathbf{x}\in X} \left(1 - p^d(\mathbf{x}) + p^d(\mathbf{x}) e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}), g-1>}\right). && \text{(4.70b)} \end{aligned}$$

3. By using the definition of the PGFL of the likelihood $F_{\mathcal{Z}_k|\mathcal{X}_k}$ and replacing its expression (4.70) into the expression for PGFL of the joint distribution $(\mathcal{Z}_k, \mathcal{X}_k)$ (4.66), we obtain

that

$$F_{(\mathcal{Z}_k,\mathcal{X}_k)}(g,h) = \int\int g^Z h^X f_{\mathcal{Z}_k|\mathcal{X}_k}(Z|X) f_{k|k-1}(X)\,\delta X\,\delta\mathbf{z} \tag{4.71a}$$

$$= \int g^Z \int h^X f_{\mathcal{Z}_k|\mathcal{X}_k}(Z|X)\,\delta X f_{k|k-1}(X)\,\delta\mathbf{z} \tag{4.71b}$$

$$= \int h^X \int g^Z f_{\mathcal{Z}_k|\mathcal{X}_k}(Z|X)\,\delta\mathbf{z}\, f_{k|k-1}(X)\,\delta X \tag{4.71c}$$

$$= \int h^X F_{\mathcal{Z}_k|\mathcal{X}_k}(g|X) f_{k|k-1}(X)\,\delta X \tag{4.71d}$$

$$= \int h^X e^{<D^c,g-1>} \prod_{\mathbf{x}\in X} \left(1 - p^d(\mathbf{x}) + p^d(\mathbf{x}) e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}),g-1>}\right) f_{k|k-1}(X)\,\delta X \tag{4.71e}$$

$$= e^{<D^c,g-1>} \int h^X \prod_{\mathbf{x}\in X} \left(1 - p^d(\mathbf{x}) + p^d(\mathbf{x}) e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}),g-1>}\right) f_{k|k-1}(X)\,\delta X. \tag{4.71f}$$

Since

$$\mathbf{x} \to 1 - p^d(\mathbf{x}) + p^d(\mathbf{x}) e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}),g-1>} \tag{4.72}$$

is a test function, we use the definition of a PGFL to obtain the expression that relates the joint PGFL $F_{(\mathcal{Z}_k,\mathcal{X}_k)}$ and the PGFL for the predicted distribution $\mathcal{X}_k$:

$$F_{(\mathcal{Z}_k,\mathcal{X}_k)}(g,h) = e^{<D^c,g-1>} \int h^X \prod_{\mathbf{x}\in X} \left(1 - p^d(\mathbf{x}) + p^d(\mathbf{x}) e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}),g-1>}\right) f_{k|k-1}(X)\,\delta X \tag{4.73a}$$

$$= e^{<D^c,g-1>} \int \left(h\left(1 - p^d + p^d e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}),g-1>}\right)\right)^X f_{k|k-1}(X)\,\delta X \tag{4.73b}$$

$$= e^{<D^c,g-1>} F_{k|k-1}\left(h\left(1 - p^d + p^d e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}),g-1>}\right)\right). \tag{4.73c}$$

4. By hypothesis the predicted distribution $\mathcal{X}_{k|k-1}$ is a PMBM with parameters

$$D^u_{k|k-1}, \{w^j_{k|k-1}, \{(r^{j,i}_{k|k-1}, f^{j,i}_{k|k-1})\}_{i\in\mathbb{I}^j_{k|k-1}}\}_{j\in\mathbb{J}_{k|k-1}}. \tag{4.74}$$

Hence, by Proposition 3.6.1, Proposition 3.6.4 and Theorem 3.6.1, the PGFL of $\mathcal{X}_{k|k-1}$ is given by

$$F_{\mathcal{X}_{k|k-1}} = F^{\text{PPP}}_{\mathcal{X}_{k|k-1}} F^{\text{MBM}}_{\mathcal{X}_{k|k-1}} \tag{4.75}$$

where

$$F^{\text{PPP}}_{\mathcal{X}_{k|k-1}}(h) = e^{<D^u_{k|k-1},h-1>} \tag{4.76a}$$

$$F^{\text{MBM}}_{\mathcal{X}_{k|k-1}}(h) = \sum_{j\in\mathbb{J}_{k|k-1}} w^j_{k|k-1} \prod_{i\in\mathbb{I}^j_{k|k-1}} F^{j,i}_{\mathcal{X}_{k|k-1}}(h) \tag{4.76b}$$

$$F^{j,i}_{\mathcal{X}_{k|k-1}}(h) = 1 - r^{j,i}_{k|k-1} + r^{j,i}_{k|k-1} < f^{j,i}_{k|k-1}, h > . \tag{4.76c}$$

By inserting the expressions in (4.75) and (4.76) into (4.73c), we can rewrite the joint PGFL $F_{(\mathcal{Z}_k,\mathcal{X}_k)}$ as

$$F_{(\mathcal{Z}_k,\mathcal{X}_k)}(g,h) = e^{<D^c,g-1>} F_{k|k-1}\left(h\left(1-p^d+p^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g-1>}\right)\right) \tag{4.77a}$$

$$= \sum_{j\in\mathbb{J}_{k|k-1}} w^j_{k|k-1} G^j(g,h)\,, \tag{4.77b}$$

where

$$G^j(g,h) = G^u(g,h) \prod_{i\in\mathbb{I}^j_{k|k-1}} G^{j,i}(h) \tag{4.78a}$$

$$G^u(g,h) = e^{<D^c,g-1>} F^{\text{PPP}}_{\mathcal{X}_{k|k-1}}\left(h\left(1-p^d+p^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g-1>}\right)\right) \tag{4.78b}$$

$$= e^{<D^c,g-1>+<D^u_{k|k-1},h\left(1-p^d+p^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g-1>}\right)-1>} \tag{4.78c}$$

$$G^{j,i}(g,h) = F^{j,i}_{\mathcal{X}_{k|k-1}}\left(h\left(1-p^d+p^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g-1>}\right)\right) \tag{4.78d}$$

$$= 1-r^{j,i}_{k|k-1}+r^{j,i}_{k|k-1} < f^{j,i}_{k|k-1}, h\left(1-p^d+p^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g-1>}\right) > . \tag{4.78e}$$

The introduced functionals G^u and $G^{j,i}$ are used to find the functional derivative of $F_{(\mathcal{Z}_k,\mathcal{X}_k)}$ in a structured manner by calculating their functional derivative and then using the results for PGFLs from Section 3.6.

5. We find the functional derivative of $G^j(g,h)$ with respect to a set of measurements Z, and evaluate it for $g(\mathbf{z}=0$.

   Since G^j is defined as the product of G^u and $G^{j,i}$ for $i\in\mathbb{I}^j_{k|k-1}$ (4.78a), it follows from the product rule for PGFls (3.6.3) that the functional derivative of G^j is given by

$$\frac{\delta G^j}{\delta Z} = \sum_{Z_0 \uplus \biguplus_{i\in\mathbb{I}^j_{k|k-1}} Z_i = Z} \frac{\delta G^u}{\delta Z_0} \prod_{i\in\mathbb{I}^j_{k|k-1}} \frac{\delta G^{j,i}}{\delta Z_i}. \tag{4.79a}$$

   The functional $G^u(g,h)$ is an exponential function with exponent

$$\ln(G^u(g,h)) =< D^c, g-1 > + < D^u_{k|k-1}, h\left(1-p^d+p^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),g-1>}\right)-1 > . \tag{4.80}$$

   Therefore, equation (4.79) can be rewritten as

$$\frac{\delta G^j}{\delta Z} = \sum_{Z_0 \uplus \biguplus_{i\in\mathbb{I}^j_{k|k-1}} Z_i = Z} \frac{\delta G^u}{\delta Z_0} \prod_{i\in\mathbb{I}^j_{k|k-1}} \frac{\delta G^{j,i}}{\delta Z_i} \tag{4.81a}$$

$$= G^u \sum_{Z_0 \uplus \biguplus_{i\in\mathbb{I}^j_{k|k-1}} Z_i = Z} \frac{\delta \ln(G^u)}{\delta Z_0} \prod_{i\in\mathbb{I}^j_{k|k-1}} \frac{\delta G^{j,i}}{\delta Z_i}. \tag{4.81b}$$

   Let us observe that the partitions of the measurement set Z used to index the summation in (4.81b), can be identified with all possible data association hypotheses.

Indeed: Given a partition of Z in Z_0 and Z_i with $i \in \mathbb{I}^j_{k|k-1}$, the subset of measurements Z_0 is associated to the background, while the sets Z_i with $i \in \mathbb{I}^j_{k|k-1}$ are associated to the object with index i. Conversely, let $A \in \mathcal{A}^j$ be a data association hypothesis for the measurement set Z, and let $C \in A$ be a cell of the data association hypothesis. If the cell C is associated to the background, i.e., $C \cap \mathbb{I}^j_{k|k-1} = \emptyset$, then we take $Z_0 = Z_C$. On the other side, if the cell C is associated to a track, i.e., $C \cap \mathbb{I}^j_{k|k-1} = \{i\}$, then we take $Z_i = Z_C$.

With this identification in mind, we apply Lemma 4.4.1 and Lemma 4.4.2, which give the functional derivatives of $\ln(G^u)$ and $G^{j,i}$, respectively. Note that for the application of Lemma 4.4.1 is it important to distinguish whether Z_0 has one or more elements, while for the application of Lemma 4.4.2 Z_i has to be non-empty. However, if Z_i is the empty set, the functional derivative with respect to Z_i, leaves $G^{j,i}$ unchanged. The application of Lemma 4.4.1 and Lemma 4.4.2 to (4.81) yields

$$\frac{\delta G^j}{\delta Z} = G^u \sum_{Z_0 \uplus \biguplus_{i \in \mathbb{I}^j_{k|k-1}} Z_i = Z} \frac{\delta \ln(G^u)}{\delta Z_0} \prod_{i \in \mathbb{I}^j_{k|k-1}} \frac{\delta G^{j,i}}{\delta Z_i} \tag{4.82a}$$

$$= G^u \sum_{A \in \mathcal{A}^j} \prod_{C \in A} G^{j,C}, \tag{4.82b}$$

where

$$G^{j,C}(g,h) = \begin{cases} D^c(\mathbf{z}_c) + \left\langle h, D^u_{k|k-1} l_{\{\mathbf{z}\}} e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}), g>} \right\rangle & \text{, if } C \cap \mathbb{I}^j = \emptyset,\ Z_C = \{\mathbf{z}_c\} \neq \emptyset \\ \left\langle D^u_{k|k-1}, h l_C e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}), g>} \right\rangle & \text{, if } C \cap \mathbb{I}^j = \emptyset,\ |Z_C| > 1 \\ G^{j,i_C}(g,h) & \text{, if } C \cap \mathbb{I}^j = \{i_C\},\ Z_C = \emptyset \\ r^{j,i_C}_{k|k-1} \left\langle h, f^{j,i_C} l_C e^{<\lambda^m(\mathbf{x}) l(\cdot|\mathbf{x}), g>} \right\rangle & \text{, if } C \cap \mathbb{I}^j = \{i_C\},\ Z_C \neq \emptyset. \end{cases} \tag{4.83}$$

6. Finally, we find the PFGL of the updated distribution $\mathcal{X}_k|\mathcal{Z}_k$, and prove that this is a PMBM with the stated parameter values.

By (4.65), (4.77b) and the linearity of the functional derivative, the PFGL of the updated distribution $\mathcal{X}_k|\mathcal{Z}_k$ is proportional to

$$F_{\mathcal{X}_k}(h|Z_k) \propto \frac{\delta}{\delta Z_k} F_{(\mathcal{Z}_k, \mathcal{X}_k)}(g,h)\Big|_{g=0} \tag{4.84a}$$

$$= \frac{\delta \sum_{j \in \mathbb{J}_{k|k-1}} w^j_{k|k-1} G^j(g,h)}{\delta Z_k}\Big|_{g=0} \tag{4.84b}$$

$$= \sum_{j \in \mathbb{J}_{k|k-1}} w^j_{k|k-1} \frac{\delta G^j(g,h)}{\delta Z_k}\Big|_{g=0}. \tag{4.84c}$$

By inserting (4.82) into (4.84) and evaluating the expression at $g = 0$, we obtain that

$$F_{\mathcal{X}_k}(h|Z_k) \propto \frac{\delta}{\delta Z_k} F_{(\mathcal{Z}_k,\mathcal{X}_k)}(g,h)\Big|_{g=0} \tag{4.85a}$$

$$= \sum_{j\in\mathbb{J}_{k|k-1}} w^j_{k|k-1} \frac{\delta G^j(g,h)}{\delta Z_k}\Big|_{g=0} \tag{4.85b}$$

$$= \sum_{j\in\mathbb{J}_{k|k-1}} w^j_{k|k-1} G^u(0,h) \sum_{A\in\mathcal{A}^j} \prod_{C\in A} G^{j,C}(0,h) \tag{4.85c}$$

$$= G^u(0,h) \sum_{j\in\mathbb{J}_{k|k-1}} w^j_{k|k-1} \sum_{A\in\mathcal{A}^j} \prod_{C\in A} G^{j,C}(0,h) \tag{4.85d}$$

We use (4.78c), (4.78e) and (4.82) to evaluate G^u, $G^{j,i}$ and $G^{j,C}$ at $g = 0$. This yields

$$G^u(0,h) = e^{<D^c,0-1>+<D^u_{k|k-1},h\left(1-p^d+p^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),0-1>}\right)-1>} \tag{4.86a}$$

$$= e^{-\lambda^c+<D^u_{k|k-1},h\left(1-p^d+p^d e^{-\lambda^m(\mathbf{x})}\right)-1>} \tag{4.86b}$$

$$= e^{-\lambda^c+<D^u_{k|k-1},hq^d-1>} \tag{4.86c}$$

$$= e^{-\lambda^c-\lambda^u_{k|k-1}+<D^u_{k|k-1}q^d,h>} \tag{4.86d}$$

$$= e^{-\lambda^c-\lambda^u_{k|k-1}+<D^u_{k|k-1}q^d,1>+<D^u_{k|k-1}q^d,h-1>}\,, \tag{4.86e}$$

$$G^{j,i}(0,h) = 1 - r^{j,i}_{k|k-1} + r^{j,i}_{k|k-1} < f^{j,i}_{k|k-1}, h\left(1-p^d+p^d e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),0-1>}\right) > \tag{4.87a}$$

$$= 1 - r^{j,i}_{k|k-1} + r^{j,i}_{k|k-1} < f^{j,i}_{k|k-1}, hq^d > \tag{4.87b}$$

$$= 1 - r^{j,i}_{k|k-1} + r^{j,i}_{k|k-1} < h, f^{j,i}_{k|k-1}q^d >\,, \tag{4.87c}$$

$$G^{j,C}(0,h) = \begin{cases} D^c(\mathbf{z}_c) + \left\langle h, D^u_{k|k-1} l_{\{\mathbf{z}\}} e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),0>}\right\rangle & \text{, if } C\cap\mathbb{I}^j = \emptyset,\ Z_C = \{\mathbf{z}_c\} \neq \emptyset \\ \left\langle D^u_{k|k-1}, h l_C e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),0>}\right\rangle & \text{, if } C\cap\mathbb{I}^j = \emptyset,\ |Z_C| > 1 \\ G^{j,i_C}(0,h) & \text{, if } C\cap\mathbb{I}^j = \{i_C\},\ Z_C = \emptyset \\ r^{j,i_C}_{k|k-1}\left\langle h, f^{j,i_C} l_C e^{<\lambda^m(\mathbf{x})l(\cdot|\mathbf{x}),0>}\right\rangle & \text{, if } C\cap\mathbb{I}^j = \{i_C\},\ Z_C \neq \emptyset \end{cases} \tag{4.88a}$$

$$= \begin{cases} D^c(\mathbf{z}_c) + \left\langle h, D^u_{k|k-1} l_{\{\mathbf{z}\}}\right\rangle & \text{, if } C\cap\mathbb{I}^j = \emptyset,\ Z_C = \{\mathbf{z}_c\} \neq \emptyset \\ \left\langle h, D^u_{k|k-1} l_C\right\rangle & \text{, if } C\cap\mathbb{I}^j = \emptyset,\ |Z_C| > 1 \\ 1 - r^{j,i_C}_{k|k-1} + r^{j,i_C}_{k|k-1} < h, f^{j,i_C}_{k|k-1} q^d > & \text{, if } C\cap\mathbb{I}^j = \{i_C\},\ Z_C = \emptyset \\ r^{j,i_C}_{k|k-1}\langle h, f^{j,i_C} l_C\rangle & \text{, if } C\cap\mathbb{I}^j = \{i_C\},\ Z_C \neq \emptyset\,, \end{cases} \tag{4.88b}$$

where we have used that $q^d(\mathbf{x}) = 1 - p^d + p^d e^{\lambda^m(\mathbf{x})}$ (4.2).

By inserting (4.86) into (4.85), and using that $-\lambda^c - \lambda^u_{k|k-1} + < D^u_{k|k-1}q^d, 1 >$ is a constant, we obtain that

$$F_{\mathcal{X}_k}(h|Z_k) \propto G^u(0,h) \sum_{j\in\mathbb{J}_{k|k-1}} w^j_{k|k-1} \sum_{A\in\mathcal{A}^j} \prod_{C\in A} G^{j,C}(0,h) \tag{4.89a}$$

$$= e^{-\lambda^c - \lambda^u_{k|k-1} + <D^u_{k|k-1} q^d, 1>} e^{<D^u_{k|k-1} q^d, h-1>} \sum_{j\in\mathbb{J}_{k|k-1}} w^j_{k|k-1} \sum_{A\in\mathcal{A}^j} \prod_{C\in A} G^{j,C}(0,h) \tag{4.89b}$$

$$\propto e^{<D^u_{k|k-1} q^d, h-1>} \sum_{j\in\mathbb{J}_{k|k-1}} w^j_{k|k-1} \sum_{A\in\mathcal{A}^j} \prod_{C\in A} G^{j,C}(0,h). \tag{4.89c}$$

We now apply Lemma 4.4.3 to each term $G^{j,C}(0,h)$ in (4.89), which allow us to rewrite (4.89) as

$$F_{\mathcal{X}_k}(h|Z_k) \propto e^{<D^u_{k|k-1} q^d, h-1>} \sum_{j\in\mathbb{J}_{k|k-1}} w^j_{k|k-1} \sum_{A\in\mathcal{A}^j} \prod_{C\in A} G^{j,C}(0,h) \tag{4.90a}$$

$$= e^{<D^u_{k|k-1} q^d, h-1>} \sum_{j\in\mathbb{J}_{k|k-1}} w^j_{k|k-1} \sum_{A\in\mathcal{A}^j} \prod_{C\in A} L^{j,C} \left(1 - r^{j,C}_k + r^{j,C}_k < h, f^{j,C}_k >\right), \tag{4.90b}$$

where

$$L^{j,C} = \begin{cases} D^c(\mathbf{z}_c) + < D^u_{k|k-1}, l_C > & \text{, if } C \cap \mathbb{I}^j = \emptyset,\ Z_C = \{\mathbf{z}_c\} \neq \emptyset \\ < D^u_{k|k-1}, l_C > & \text{, if } C \cap \mathbb{I}^j = \emptyset,\ |Z_C| > 1 \\ 1 - r^{j,i_C}_{k|k-1} + r^{j,i_C}_{k|k-1} < f^{j,i_C}_{k|k-1}, q^d > & \text{, if } C \cap \mathbb{I}^j = \{i_C\},\ Z_C = \emptyset \\ r^{j,i_C}_{k|k-1} < f^{j,i_C}_{k|k-1}, q^d > & \text{, if } C \cap \mathbb{I}^j = \{i_C\},\ Z_C \neq \emptyset, \end{cases} \tag{4.91a}$$

$$r^{j,C}_k = \begin{cases} \frac{<D^u_{k|k-1}, l_C>}{D^c(\mathbf{z}_c) + <D^u_{k|k-1}, l_C>} & \text{, if } C \cap \mathbb{I}^j = \emptyset,\ Z_C = \{\mathbf{z}_c\} \neq \emptyset \\ 1 & \text{, if } C \cap \mathbb{I}^j = \emptyset,\ |Z_C| > 1 \\ \frac{r^{j,i_C}_{k|k-1} <f^{j,i_C}_{k|k-1}, q^d>}{1 - r^{j,i_C}_{k|k-1} + r^{j,i_C}_{k|k-1} <f^{j,i_C}_{k|k-1}, q^d>} & \text{, if } C \cap \mathbb{I}^j = \{i_C\},\ Z_C = \emptyset \\ 1 & \text{, if } C \cap \mathbb{I}^j = \{i_C\},\ Z_C \neq \emptyset, \end{cases} \tag{4.91b}$$

$$f^{j,C}_k(\mathbf{x}) = \begin{cases} \frac{l_C(\mathbf{x}) D^u_{k|k-1}(\mathbf{x})}{<l_C, D^u_{k|k-1}>} & \text{, if } C \cap \mathbb{I}^j = \emptyset \\ \frac{q^d(\mathbf{x}) f^{j,i_C}_{k|k-1}(\mathbf{x})}{<q^d, f^{j,i_C}_{k|k-1}>} & \text{, if } C \cap \mathbb{I}^j = \{i_C\},\ Z_C = \emptyset \\ \frac{l_C(\mathbf{x}) f^{j,i_C}_{k|k-1}(\mathbf{x})}{<l_C, f^{j,i_C}_{k|k-1}>} & \text{, if } C \cap \mathbb{I}^j = \{i_C\},\ Z_C \neq \emptyset. \end{cases} \tag{4.91c}$$

The functional in (4.90b) has the structure of the PGFL of a PMBM (4.12) with the required parameter expressions, except for its weights, which are not necessarily normalized. Since this functional is proportional to the PGFL of the posterior distribution $\mathcal{X}_k$, we conclude this PGFL is the functional in (4.90b), but with normalized weights. The expression for the normalized weights is

$$w^{j,A}_k = \frac{w^j_{k|k-1} \prod_{C\in A} L^{j,C}}{\sum_{j\in\mathbb{J}_{k|k-1}} \sum_{A\in\mathcal{A}^j} w^j_{k|k-1} \prod_{C\in A} L^{j,C}}, \tag{4.92}$$

as we wanted to prove. □

4.4.1 Special case of prediction and update steps

Theorem 4.4.1 and Theorem 4.4.2 give general formulas for the parameters of the PMBM distribution (4.13). However, in the general case, it is not possible to derive closed-form expressions for these parameters, and one would have to resort to numerical methods, such as numerical integration, in order to get estimates of the PMBM parameters after the prediction and update steps.

In the following two theorems, we show that by adding extra hypotheses to Theorem 4.4.1 and Theorem 4.4.2 closed-form expressions can be derived for the PMBM parameters.

Theorem 4.4.3. *(PMBM prediction step for normal case) In addition to the hypotheses of Theorem 4.4.1, assume that the survival probability p^s is constant, that the intensity of the PPP for the unknown objects D_k^u is a linear combination of normal probability density functions, i.e.,*

$$D_k^u(\mathbf{x}) = \sum_{n=1}^{N^u} d_n^u \mathcal{N}\left(\mathbf{x}, \mathbf{x}_n^u, \mathbf{P}_n^u\right) , \tag{4.93}$$

that the spatial distribution of the Bernoulli components is normally distributed as

$$f_k^{j,i}(\mathbf{x}) = \mathcal{N}\left(\mathbf{x}, \mathbf{x}^{j,i}, \mathbf{P}^{j,i}\right) , \tag{4.94}$$

and that the state-transition is also normally distributed as

$$f_k(\mathbf{x}|\mathbf{x}') = \mathcal{N}\left(\mathbf{x}, \mathbf{F}\mathbf{x}', \mathbf{Q}\right) . \tag{4.95}$$

Then, the parameters of the updated PMBM distribution are equal to

$$D_{k|k-1}^u(\mathbf{x}) = D^b(\mathbf{x}) + p^s \sum_{n=1}^{N^u} d_n^u \mathcal{N}\left(\mathbf{x}, \mathbf{F}\mathbf{x}_n^u, \mathbf{F}\mathbf{P}_n^u\mathbf{F}^T + \mathbf{Q}\right) \tag{4.96a}$$

$$w_{k|k-1}^j = w_k^j \tag{4.96b}$$

$$r_{k|k-1}^{j,i} = r^{j,i} p^s \tag{4.96c}$$

$$f_{k|k-1}^{j,i}(\mathbf{x}) = \mathcal{N}\left(\mathbf{x}, \mathbf{F}\mathbf{x}^{j,i}, \mathbf{F}\mathbf{P}^{i,j}\mathbf{F}^T + \mathbf{Q}\right) . \tag{4.96d}$$

In particular, if the intensity of the PPP for the generation of new objects $D^b(\mathbf{x})$ is also a linear combination of normal probability density functions, then the intensity of the updated PPP for the unknown objects has also the same structure.

Proof. From Theorem 4.4.1, we know that

$$D_{k|k-1}^u(\mathbf{x}) = D^b(\mathbf{x}) + < D_k, p^s f_k(\mathbf{x}|\cdot) > \tag{4.97a}$$

$$w_{k|k-1}^j = w_k^j \tag{4.97b}$$

$$r_{k|k-1}^{j,i} = r^{j,i} < f_k^{j,i}, p^s > \tag{4.97c}$$

$$f_{k|k-1}^{j,i}(\mathbf{x}) = \frac{< f_k^{j,i}, p^s f_k(\mathbf{x}|\cdot) >}{< f_k^{j,i}, p^s >} \tag{4.97d}$$

In particular, we already have that $w_{k|k-1}^j = w_k^j$.

Furthermore, p^s is constant and $f^{j,i}$ integrates up to 1 because is a probability density function. Therefore,

$$r^{j,i}_{k|k-1} = r^{j,i} < f^{j,i}_k, p^s > \tag{4.98a}$$

$$= r^{j,i} p^s < f^{j,i}_k, 1 >= r^{j,i} p^s. \tag{4.98b}$$

By the same argument, we have that

$$f^{j,i}_{k|k-1}(\mathbf{x}) = \frac{< f^{j,i}_k, p^s f_k(\mathbf{x}|\cdot) >}{< f^{j,i}_k, p^s >} \tag{4.99a}$$

$$= \frac{p^s < f^{j,i}_k, f_k(\mathbf{x}|\cdot) >}{p^s < f^{j,i}_k, 1 >} \tag{4.99b}$$

$$=< f^{j,i}_k, f_k(\mathbf{x}|\cdot) > . \tag{4.99c}$$

Now, by Theorem A.2 from Appendix A, the product of the probability density functions $f^{j,i}_k(\mathbf{x}')$ and $f_k(\mathbf{x}|\mathbf{x}')$ is equal to

$$\mathcal{N}\left(\begin{bmatrix}\mathbf{x}'\\ \mathbf{x}\end{bmatrix}, \begin{bmatrix}\mathbf{x}^{j,i}\\ \mathbf{F}\mathbf{x}^{j,i}\end{bmatrix}, \begin{bmatrix}\mathbf{P}^{j,i} & \mathbf{P}^{j,i}\mathbf{F}\\ \mathbf{F}\mathbf{P}^{j,i} & \mathbf{F}\mathbf{P}^{j,i}\mathbf{F}^T + \mathbf{Q}\end{bmatrix}\right). \tag{4.100}$$

Hence, the integral $< f^{j,i}_k, f_k(\mathbf{x}|\cdot) >$ gives the marginal distribution of $\mathbf{x}$. Therefore

$$f^{j,i}_{k|k-1}(\mathbf{x}) =< f^{j,i}_k, f_k(\mathbf{x}|\cdot) >= \mathcal{N}\left(\mathbf{x}, \mathbf{F}\mathbf{x}^{j,i}, \mathbf{F}\mathbf{P}^{i,j}\mathbf{F}^T + \mathbf{Q}\right). \tag{4.101}$$

By applying Theorem A.2 from Appendix A and using an analogous argument, we find the expression for the birth intensity:

$$D^u_{k|k-1}(\mathbf{x}) = D^b(\mathbf{x}) + < D_k, p^s f_k(\mathbf{x}|\cdot) > \tag{4.102a}$$

$$= D^b(\mathbf{x}) + p^s < \sum_{n=1}^{N^u} d^u_n \mathcal{N}\left(\cdot, \mathbf{x}^u_n, \mathbf{P}^u_n\right), f_k(\mathbf{x}|\cdot) > \tag{4.102b}$$

$$= D^b(\mathbf{x}) + p^s \sum_{n=1}^{N^u} d^u_n < \mathcal{N}\left(\cdot, \mathbf{x}^u_n, \mathbf{P}^u_n\right), f_k(\mathbf{x}|\cdot) > \tag{4.102c}$$

$$D^b(\mathbf{x}) + p^s \sum_{n=1}^{N^u} d^u_n \mathcal{N}\left(\mathbf{x}, \mathbf{F}\mathbf{x}^u_n, \mathbf{F}\mathbf{P}^u_n\mathbf{F}^T + \mathbf{Q}\right), \tag{4.102d}$$

which completes the proof. □

Theorem 4.4.4. *(PMBM update step for normal case) In addition to the hypotheses of Theorem 4.4.2, assume that the detection probability p^d is constant, that the Poisson rate of the measurements λ^m is constant, that the intensity of the PPP for the unknown objects $D^u_{k|k-1}$ is a linear combination of normal probability density functions, i.e.,*

$$D^u_{k|k-1}(\mathbf{x}) = \sum_{n=1}^{N^u} d^u_n \mathcal{N}\left(\mathbf{x}, \mathbf{x}^u_n, \mathbf{P}^u_n\right), \tag{4.103}$$

that the spatial distribution of the Bernoulli components is normally distributed as

$$f_{k|k-1}^{j,i}(\mathbf{x}) = \mathcal{N}\left(\mathbf{x}, \mathbf{x}^{j,i}, \mathbf{P}^{j,i}\right) , \tag{4.104}$$

and that the likelihood is also normally distributed as

$$l(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\mathbf{z}, \mathbf{H}\mathbf{x}, \mathbf{R}\right) . \tag{4.105}$$

Then the updated intensity of the PPP for the unknown objects is

$$D_k^u(\mathbf{x}) = q^d D_{k|k-1}^u(\mathbf{x}) = q^d \sum_{n=1}^{N^u} d_n^u \mathcal{N}\left(\mathbf{x}, \mathbf{x}_n^u, \mathbf{P}_n^u\right) , \tag{4.106}$$

the weights of the MBM for the tracked objects are

$$w_k^{j,A} = \frac{w_{k|k-1}^j \prod_{C\in A} L^{j,C}}{\sum_{j\in\mathbb{J}_{k|k-1}} \sum_{A\in\mathcal{A}^j} w_{k|k-1}^j \prod_{C\in A} L^{j,C}} , \tag{4.107a}$$

$$L^{j,C} = \begin{cases} D^c(\mathbf{z}_c) + < D_{k|k-1}^u, l_C > & , \textit{if } C \cap \mathbb{I}^j = \emptyset,\ Z_C = \{\mathbf{z}_c\} \neq \emptyset \\ < D_{k|k-1}^u, l_C > & , \textit{if } C \cap \mathbb{I}^j = \emptyset,\ |Z_C| > 1 \\ 1 - r_{k|k-1}^{j,i_C} + r_{k|k-1}^{j,i_C} q^d & , \textit{if } C \cap \mathbb{I}^j = \{i_C\},\ Z_C = \emptyset \\ r_{k|k-1}^{j,i_C} q^d & , \textit{if } C \cap \mathbb{I}^j = \{i_C\},\ Z_C \neq \emptyset , \end{cases} \tag{4.107b}$$

and the parameters of the Bernoulli components are

$$r_k^{j,C} = \begin{cases} \frac{< D_{k|k-1}^u, l_C >}{D^c(\mathbf{z}_c) + < D_{k|k-1}^u, l_C >} & , \textit{if } C \cap \mathbb{I}^j = \emptyset,\ Z_C = \{\mathbf{z}_c\} \neq \emptyset \\ 1 & , \textit{if } C \cap \mathbb{I}^j = \emptyset,\ |Z_C| > 1 \\ \frac{r_{k|k-1}^{j,i_C} q^d}{1 - r_{k|k-1}^{j,i_C} + r_{k|k-1}^{j,i_C} q^d} & , \textit{if } C \cap \mathbb{I}^j = \{i_C\},\ Z_C = \emptyset \\ 1 & , \textit{if } C \cap \mathbb{I}^j = \{i_C\},\ Z_C \neq \emptyset , \end{cases} \tag{4.108a}$$

$$f_k^{j,C}(\mathbf{x}) = \begin{cases} \frac{l_C(\mathbf{x}) D_{k|k-1}^u(\mathbf{x})}{< l_C, D_{k|k-1}^u >} & , \textit{if } C \cap \mathbb{I}^j = \emptyset \\ f_{k|k-1}^{j,i_C}(\mathbf{x}) & , \textit{if } C \cap \mathbb{I}^j = \{i_C\},\ Z_C = \emptyset \\ \frac{l_C(\mathbf{x}) f_{k|k-1}^{j,i_C}(\mathbf{x})}{< l_C, f_{k|k-1}^{j,i_C} >} & , \textit{if } C \cap \mathbb{I}^j = \{i_C\},\ Z_C \neq \emptyset , \end{cases} \tag{4.108b}$$

where

$$q^d = 1 - p^d + p^d e^{-\lambda^m} \tag{4.109a}$$

$$D^u_{k|k-1}(\mathbf{x}) l_C(\mathbf{x}) = p^d e^{-\lambda^m} \lambda^{m|Z_C|} \sum_{n=1}^{N^u} d^u_n \times \ldots$$

$$\ldots \mathcal{N}\left(\begin{bmatrix} \mathbf{x} \\ \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_m \end{bmatrix}, \begin{bmatrix} \mathbf{x}^u_n \\ \mathbf{H}\mathbf{x}^u_n \\ \vdots \\ \mathbf{H}\mathbf{x}^u_n \end{bmatrix}, \begin{bmatrix} \mathbf{P}^u_n & \mathbf{P}^u_n\mathbf{H}^T & \mathbf{P}^u_n\mathbf{H}^T & \ldots & \ldots & \mathbf{P}^u_n\mathbf{H}^T \\ \mathbf{H}\mathbf{P}^u_n & \mathbf{H}\mathbf{P}^u_n\mathbf{H}^T + \mathbf{R} & \mathbf{0} & \ldots & \ldots & \mathbf{0} \\ \mathbf{H}\mathbf{P}^u_n & \mathbf{0} & \mathbf{H}\mathbf{P}^u_n\mathbf{H}^T + \mathbf{R} & \mathbf{0} & \ddots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ \mathbf{H}\mathbf{P}^u_n & \mathbf{0} & \ldots & \ldots & \mathbf{0} & \mathbf{H}\mathbf{P}^u_n\mathbf{H}^T + \mathbf{R} \end{bmatrix} \right) \tag{4.109b}$$

$$< D^u_{k|k-1}, l_C >= p^d e^{-\lambda^m} \lambda^{m|Z_C|} \sum_{n=1}^{N^u} d^u_n \prod_{\mathbf{z} \in Z_C} \mathcal{N}\left(\mathbf{z}, \mathbf{H}\mathbf{x}^u_n, \mathbf{H}\mathbf{P}^u_n\mathbf{H}^T + \mathbf{R}\right) \tag{4.109c}$$

$$f^{j,i_C}_{k|k-1}(\mathbf{x}) l_C(\mathbf{x}) = e^{-\lambda^m} \lambda^{m|Z_C|} \times \ldots$$

$$\ldots \mathcal{N}\left(\begin{bmatrix} \mathbf{x} \\ \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_m \end{bmatrix}, \begin{bmatrix} \mathbf{x}^{j,i} \\ \mathbf{H}\mathbf{x}^{j,i} \\ \vdots \\ \mathbf{H}\mathbf{x}^{j,i} \end{bmatrix}, \begin{bmatrix} \mathbf{P}^{j,i} & \mathbf{P}^{j,i}\mathbf{H}^T & \mathbf{P}^{j,i}\mathbf{H}^T & \ldots & \ldots & \mathbf{P}^{j,i}\mathbf{H}^T \\ \mathbf{H}\mathbf{P}^{j,i} & \mathbf{H}\mathbf{P}^{j,i}\mathbf{H}^T + \mathbf{R} & \mathbf{0} & \ldots & \ldots & \mathbf{0} \\ \mathbf{H}\mathbf{P}^{j,i} & \mathbf{0} & \mathbf{H}\mathbf{P}^{j,i}\mathbf{H}^T + \mathbf{R} & \mathbf{0} & \ddots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ \mathbf{H}\mathbf{P}^{j,i} & \mathbf{0} & \ldots & \ldots & \mathbf{0} & \mathbf{H}\mathbf{P}^{j,i}\mathbf{H}^T + \mathbf{R} \end{bmatrix} \right) \tag{4.109d}$$

$$< f^{j,i_C}_{k|k-1}, l_C >= e^{-\lambda^m} \lambda^{m|Z_C|} \prod_{\mathbf{z} \in Z_C} \mathcal{N}\left(\mathbf{z}, \mathbf{H}\mathbf{x}^{j,i}, \mathbf{H}\mathbf{P}^{j,i}\mathbf{H}^T + \mathbf{R}\right). \tag{4.109e}$$

Proof. Since the Poisson rate of the measurements λ^m is constant, the miss-detection probability q^d is also constant and equal to

$$q^d = 1 - p^d + p^d e^{-\lambda^m}. \tag{4.110}$$

Most of the identities of these theorem follow from the identities of Theorem 4.4.2 and the facts that q^d is constant and that probability density functions integrate up to 1.

The identities for the products $D^u_{k|k-1} l_C$ and $l_C f^{j,i_C}_{k|k-1}$ require more work. We prove only the identities for $D^u_{k|k-1} l_C$ since the ones for $l_C f^{j,i_C}_{k|k-1}$ are analogous.

By the definition of l_C (4.8) and the hypothesis about D^u, we have that

$$D^u_{k|k-1}(\mathbf{x}) l_C(\mathbf{x}) = \left(\sum_{n=1}^{N^u} d^u_n \mathcal{N}\left(\mathbf{x}, \mathbf{x}^u_n, \mathbf{P}^u_n\right) \right) p^d e^{-\lambda^m} \prod_{\mathbf{z} \in Z_C} \lambda^m \mathcal{N}\left(\mathbf{z}, \mathbf{H}\mathbf{x}, \mathbf{R}\right) \tag{4.111a}$$

$$= p^d e^{-\lambda^m} \lambda^{m|Z_C|} \sum_{n=1}^{N^u} d^u_n \mathcal{N}\left(\mathbf{x}, \mathbf{x}^u_n, \mathbf{P}^u_n\right) \prod_{\mathbf{z} \in Z_C} \mathcal{N}\left(\mathbf{z}, \mathbf{H}\mathbf{x}, \mathbf{R}\right). \tag{4.111b}$$

By Theorem A.2 in Appendix A, the product of the probability density functions in (4.111b) can be rewritten as

$$D^u_{k|k-1}(\mathbf{x})l_C(\mathbf{x}) = p^d e^{-\lambda^m}\lambda^{m|Z_C|}\sum_{n=1}^{N^u} d^u_n \times \ldots$$
$$\ldots\mathcal{N}\left(\begin{bmatrix}\mathbf{x}\\ \mathbf{z}_1\\ \vdots\\ \mathbf{z}_m\end{bmatrix}, \begin{bmatrix}\mathbf{x}^u_n\\ \mathbf{H}\mathbf{x}^u_n\\ \vdots\\ \mathbf{H}\mathbf{x}^u_n\end{bmatrix}, \begin{bmatrix}\mathbf{P}^u_n & \mathbf{P}^u_n\mathbf{H}^T & \mathbf{P}^u_n\mathbf{H}^T & \ldots & \ldots & \mathbf{P}^u_n\mathbf{H}^T\\ \mathbf{H}\mathbf{P}^u_n & \mathbf{H}\mathbf{P}^u_n\mathbf{H}^T+\mathbf{R} & \mathbf{0} & \ldots & \ldots & \mathbf{0}\\ \mathbf{H}\mathbf{P}^u_n & \mathbf{0} & \mathbf{H}\mathbf{P}^u_n\mathbf{H}^T+\mathbf{R} & \mathbf{0} & \ddots & \mathbf{0}\\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots\\ \vdots & \vdots & \ddots & \ddots & \ddots & \mathbf{0}\\ \mathbf{H}\mathbf{P}^u_n & \mathbf{0} & \ldots & \ldots & \mathbf{0} & \mathbf{H}\mathbf{P}^u_n\mathbf{H}^T+\mathbf{R}\end{bmatrix}\right), \tag{4.112}$$

where $Z_C = \{\mathbf{z}_1, \ldots, \mathbf{z}_m\}$.

In particular, the integral $< D^u_{k|k-1}, l_C >$ corresponds to the marginalization of each distribution in (4.112) with respect to the measurements, which gives

$$< D^u_{k|k-1}, l_C > = p^d e^{-\lambda^m}\lambda^{m|Z_C|}\sum_{n=1}^{N^u} d^u_n \times \ldots$$
$$\ldots\mathcal{N}\left(\begin{bmatrix}\mathbf{z}_1\\ \vdots\\ \mathbf{z}_m\end{bmatrix}, \begin{bmatrix}\mathbf{H}\mathbf{x}^u_n\\ \vdots\\ \mathbf{H}\mathbf{x}^u_n\end{bmatrix}, \begin{bmatrix}\mathbf{H}\mathbf{P}^u_n\mathbf{H}^T+\mathbf{R} & \mathbf{0} & \ldots & \ldots & \mathbf{0}\\ \mathbf{0} & \mathbf{H}\mathbf{P}^u_n\mathbf{H}^T+\mathbf{R} & \mathbf{0} & \ddots & \mathbf{0}\\ \vdots & \ddots & \ddots & \ddots & \vdots\\ \vdots & \ddots & \ddots & \ddots & \mathbf{0}\\ \mathbf{0} & \ldots & \ldots & \mathbf{0} & \mathbf{H}\mathbf{P}^u_n\mathbf{H}^T+\mathbf{R}\end{bmatrix}\right) \tag{4.113}$$
$$= p^d e^{-\lambda^m}\lambda^{m|Z_C|}\sum_{n=1}^{N^u} d^u_n \prod_{\mathbf{z}\in Z_C}\mathcal{N}\left(\mathbf{z}, \mathbf{H}\mathbf{x}^u_n, \mathbf{H}\mathbf{P}^u_n\mathbf{H}^T+\mathbf{R}\right), \tag{4.114}$$

which concludes the proof. □

We observe in Theorem 4.4.4 that the spatial distribution of the Bernoulli components, $f^{j,i}_k$, is not necessarily normally distributed, but this probability density function is in general a linear combination of normally distributed probability density functions, i.e., $f^{j,i}_k$ is a Gaussian mixture. Therefore, we assume that $f^{j,i}_k$ can be approximated by a normally distributed probability function with the same mean value and covariance matrix as the original $f^{j,i}_k$. This is a special case of moment matching, and it is a well-used approximation in OT. The equations for this approximation can be found in Theorem A.3 in Appendix A.

By using this approximation and under the hypotheses of Theorem 4.4.3 and Theorem 4.4.4, we can conclude that as long as the intensity of the birth PPP, $D^b(\mathbf{x})$, is given by a linear combination of normally distributed probability density function, and as long as the initial predicted spatial distribution of the Bernoulli components, $f^{j,i}_{1|0}$, are normally distributed, then the PMBM prediction and update step will always give an intensity of the undetected targets that is a linear

combination of normally distributed probability density functions, and Bernoulli components whose spatial distribution is normally distributed.

This invariant structure together with the closed-form expressions in Theorem 4.4.3, Theorem 4.4.4 and Theorem A.3, allow us to construct data structures for the PMBM parameters when implementing this filter.

Finally, we observe that if the state-transition probability density function $f_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$ or the likelihood $l(\mathbf{z}|\mathbf{x})$ are not linear, i.e., $f_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1}) \sim \mathcal{N}(f(\mathbf{x}_{k-1}), Q)$ or $l(\mathbf{z}|\mathbf{x}) \sim \mathcal{N}(h(\mathbf{x}), R)$, then the matrices $\mathbf{F}$ and $\mathbf{H}$ are obtained by linearizing f and h at the corresponding predicted state vectors, similarly to how it is done in an EKF.

4.5 Implementation

As shown in Section 4.4, the prediction step of the PMBM filter (Theorem 4.4.1) keeps the number of components of the MBM constant, as well as the number of Bernoulli components in each MB. However, the update step (Theorem 4.4.2) creates a MB component for each pair of predicted MB and possible association given the measurements. In other words, all possible global hypotheses that could explain how the tracked objects are distributed given the measurements, are been considered at each time step. Therefore, it should come as no surprise, that a direct implementation of the prediction and update steps for most target tracking problems is not feasible in practice due to storage limitations, as well as computational time, specially in a real-time setting.

In fact, as shown in [17], the complexity of the update step is between $\mathcal{O}\left(2^{|Z|+|\mathbb{I}^j|}\right)$ and $\mathcal{O}\left((|Z|+|\mathbb{I}^j|)!\right)$, where Z is the set of measurements available in the update step and $\mathbb{I}^j$ is the index set of the global hypotheses, i.e., the index set of the MB components.

There are diverse strategies that reduce the complexity of the update step in order to make the implementation of the PMBM filter for MEOT feasible. In the following subsections, we briefly explain the methods used in the PMBM implementation for this thesis.

4.5.1 Gating

Gating is the process of only considering data association hypotheses, whose cells have likelihoods with respect to the predicted distribution that are above a threshold. In other words, data association hypotheses that have cells whose measurements are highly unlikely given the predicted distributed, are discarded.

Gating is a well-used method for reducing the complexity of OT update steps. Although there is the theoretical possibility that a correct data association hypothesis may be discarded, the complexity reduction achieved with gating outweighs this highly unlikely possibility.

95% to 99% confidence regions for gating are usual selections.

4.5.2 M best assignments

After gating, Murty's method [29], also know as the auction algorithm, is used to find the M best data association hypotheses in the sense that the sum of the likelihood of their cells are maximized.

Values of M of the order 100 to 1000 are usual selections.

4.5.3 Pruning

Pruning is the process of eliminating Bernoulli components whose existence probabilities, $r^{j,i}$, are below a threshold, as well as the process of eliminating whole global data association hypotheses whose probabilities, w^j, are below a threshold.

Pruning is performed after finding the M best assignments, which can also be seen as a type of pruning.

Usual values for the pruning thresholds are in the order of 0.01.

Chapter 5

Gaussian Process fundamentals

A Gaussian process (GP) models the stochastic properties of an unknown function whose values are normally distributed with respect to each other. There are two main reasons that make GPes appealing for EOT. First, GPes are stochsatic models, and can therefore be smoothly merged with probabilistic filters, which are extensively used in EOT, such as the PMBM filter used in this thesis (see Chapter 4). Second, GPes are non-parametric function models, i.e., they do not impose any underlying structure on the unknown modeled functions. Therefore, the same GP can model a wide variety of functions. However, it is worth noticing that GPes depend on a set of parameters, called hyperparameters, which determine how function values are assumed to be distributed and correlated to each other.

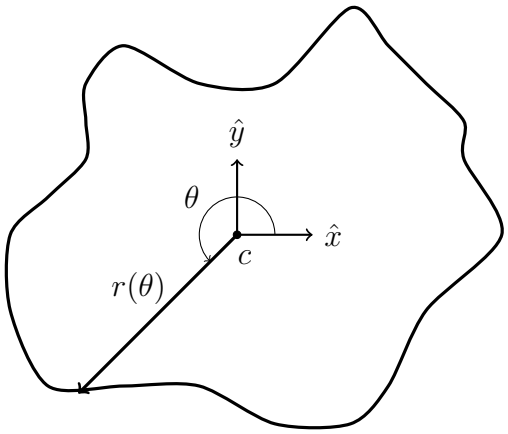


Figure 5.1: Polar parametrization of the boundary of an extend with respect to an internal point c and a local attached reference frame $(\hat{x}, \hat{y})$. The radius function $r(\theta)$ characterizes the extent.

In the context of this thesis, GPes are used to model the unknown extents of the tracked objects. More precisely, each of these processes models an unknown radius function, which corresponds to the polar parametrization of the boundary of their respective extent with respect to a point inside the extent, as illustrated in Figure 5.1. Because of the nature of this unknown radius function, we are particularly interested in GPes that can easily model non-negative and 2π-periodic functions. Furthermore, since the extents of interest represent the hulls of maritime vessels in the horizontal plane, which are usually symmetric with respect to their longitudinal axis, we are also interested in GPes that can in addition model even radius functions.

This chapter presents the GP concepts and results that are necessary for modeling the state-transition of the extent states, which is done in Section 6.2.1. For a comprehensive presentation and discussion of GPes, we highly recommend the book [30].

5.1 Definitions and examples

Definition 5.1.1 (Gaussian Process)**.** *Let $\mathbb{I}$ be an non-empty index set. A real process* [1] *$f : \mathbb{I} \to \mathbb{R}$ is said to be a Gaussian process (GP) with mean function $m : \mathbb{I} \to \mathbb{R}$ and covariance function $k : \mathbb{I} \times \mathbb{I} \to \mathbb{R}$ if any finite subset of different stochastic variables $\{f(i_1), f(i_2) \cdots, f(i_N)\}$ with $i_1, i_2, \ldots, i_N$ has a joint multivariate Gaussian distribution given by*

$$\begin{bmatrix} f(i_1) \\ f(i_2) \\ \vdots \\ f(i_N) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m(i_1) \\ m(i_2) \\ \vdots \\ m(i_N) \end{bmatrix}, \begin{bmatrix} k(i_1,i_1) & k(i_1,i_2) & \cdots & k(i_1,i_N) \\ k(i_2,i_2) & k(i_2,i_2) & \cdots & k(i_2,i_N) \\ \vdots & \vdots & \ddots & \vdots \\ k(i_N,i_1) & k(i_N,i_2) & \cdots & k(i_N,i_N) \end{bmatrix} \right). \tag{5.1}$$

In such cases, we write $f \sim \mathcal{GP}(m, k)$.

Furthermore, we introduce the following convenient GP notation:

Definition 5.1.2 (Mean vector and covariance matrix)**.** *Let $\mathcal{GP}(m, k)$ be a GP, and let $\mathbf{i} = [i_1, i_2, \cdots, i_N]^T \in \mathbb{I}^N$ and $\mathbf{j} = [j_1, j_2, \cdots, i_M]^T \in \mathbb{I}^M$ be two vectors of indices.*

We define the associated mean vector to $\mathbf{i}$, $\mathbf{M}(\mathbf{i})$, and the associated covariance matrix to $\mathbf{i}$ and $\mathbf{j}$, $\mathbf{K}(\mathbf{i}, \mathbf{j})$, as

$$\mathbf{M}(\mathbf{i}) = \begin{bmatrix} m(i_1) \\ m(i_2) \\ \vdots \\ m(i_N) \end{bmatrix} \tag{5.2a}$$

$$\mathbf{K}(\mathbf{i}, \mathbf{j}) = \begin{bmatrix} k(i_1,j_1) & k(i_1,j_2) & \cdots & k(i_1,j_M) \\ k(i_2,j_2) & k(i_2,j_2) & \cdots & k(i_2,j_M) \\ \vdots & \vdots & \ddots & \vdots \\ k(i_N,j_1) & k(i_N,j_2) & \cdots & k(i_N,j_M) \end{bmatrix}. \tag{5.2b}$$

As mentioned in the introduction to this chapter, the unknown functions modeled by the GPes correspond to the radius function of a polar parametrization. Therefore, instead of $f(x) \sim \mathcal{GP}(m(x), k(x, x'))$ with $x, x' \in \mathbb{I}$, we will rather write $r(\theta) \sim \mathcal{GP}(m(\theta), k(\theta, \theta'))$ with $\theta, \theta' \in [0, 2\pi]$ in order to reflect the nature of the unknown function.

[1] A real process is a real valued family of random variables. In this case, the family is $\{f(i)\}_{i \in \mathbb{I}}$.

5.1.1 Gaussian processes with squared exponential covariance functions

The squared exponential covariance function with bias and noise term is defined as

$$k_e(\theta, \theta') = \sigma_f^2 e^{-\frac{(\theta-\theta')^2}{2l^2}} + \sigma_b^2 + \sigma_n^2 \delta(\theta, \theta')\,, \tag{5.3}$$

where the hyperparameters $\sigma_f \geq 0$, $\sigma_b \geq 0$ and $\sigma_n \geq 0$ are called the variance of the signal amplitude, the variance of the bias and the variance of the noise, respectively, while the hyperparameter $l > 0$ is called the length-scale.

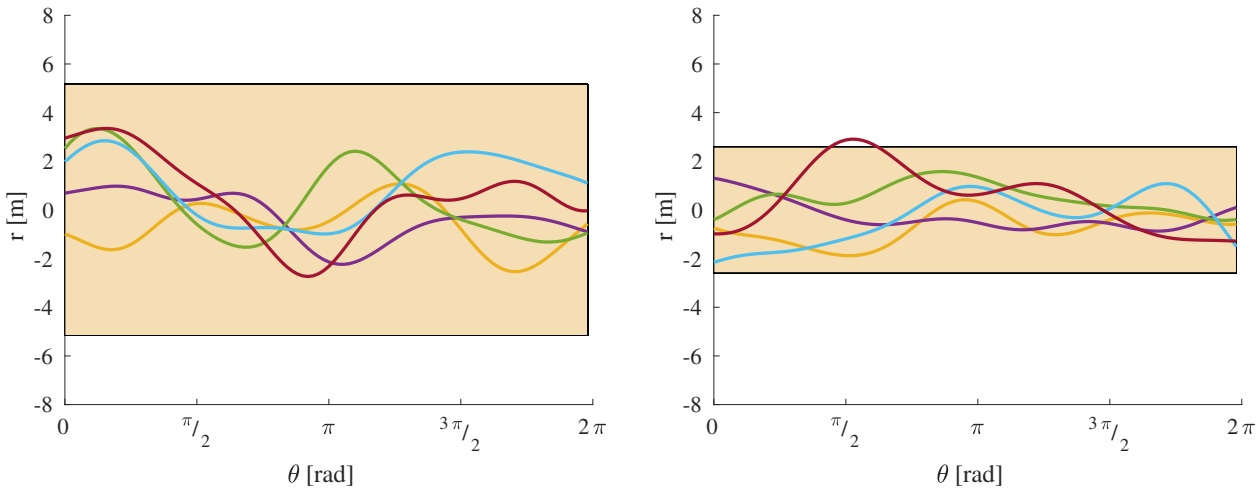
GPes with identically zero mean functions and a squared exponential covariance function with bias and noise term as their covariance functions, i.e., GPes of the form $r(\theta) \sim \mathcal{GP}(0, k_e(\theta, \theta'))$, with $\theta, \theta' \in [0, 2\pi]$, constitute one of the most well-used GP classes, and they are the basis for the other GP classes presented in this chapter.

Before we introduce these other GP classes, let us study how the hyperparameters σ_f, σ_b, σ_b and l influence the type of functions that can be modeled by $\mathcal{GP}(0, k_e)$:

Figure 5.2 illustrates what type of functions can be modeled by the GP class $\mathcal{GP}(0, k_e)$. In this example, the interval $[0, 2\pi)$ was regularly discretized using 100 angles θ_n, i.e., $\theta_n = \frac{2\pi}{N}(n-1)$ with $N = 100$ and $n = 1, \ldots, N$. Then, five functions were drawn from the resulting multivariate Gaussian distribution $\mathcal{N}(0, \mathbf{K}_e(\boldsymbol{\theta}, \boldsymbol{\theta}))$, where $\boldsymbol{\theta} = [\theta_n]_{n=1}^N$. This was done for different hyperparameter sets.

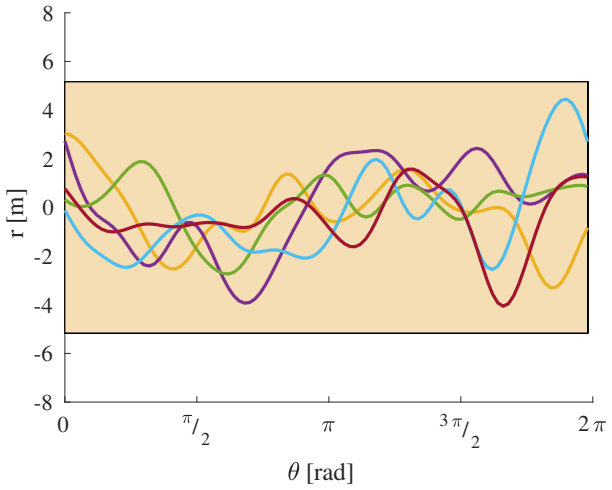
As it can be deduced from the expression in (5.3), the hyperparameter σ_f determines together with σ_b and σ_n how large the variance and covariance of different function values are, while the length-scale l models how far points have to be so that their function values are uncorrelated. Therefore, the larger the variance of the signal amplitude σ_f is, the more individual function values can vary, while the larger the length-scale l is, the more the covariance between points increases, and the function values are more likely to be slowly varying. These facts can be observed in Figures 5.2a to 5.2c.

Furthermore, we observe in Figures 5.2a to 5.2c that the mean of the drawn functions is very close to zero. Such a feature is undesirable for a radius function, which only takes non-negative values, and this can be mended with a positive σ_b hyperparameter value. The reason behind this is that a positive σ_b will add to each drawn function a random bias term r_0, which is distributed as $r_0 \in \mathcal{N}(0, \sigma_b^2)$. This fact is shown in Figures 5.2d to 5.2e. Here, we can observe that the drawn functions have a wide variety of mean values, and that the larger the variance of the bias σ_b is, the more likely larger mean values become.

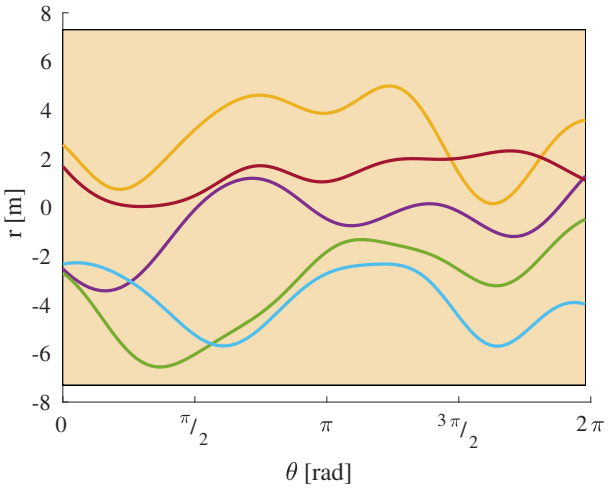


(a) $(\sigma_f, \sigma_b, \sigma_n, l) = \left(2\,\mathrm{m}, 0\,\mathrm{m}, 0\,\mathrm{m}, \frac{\pi}{4}\right)$.

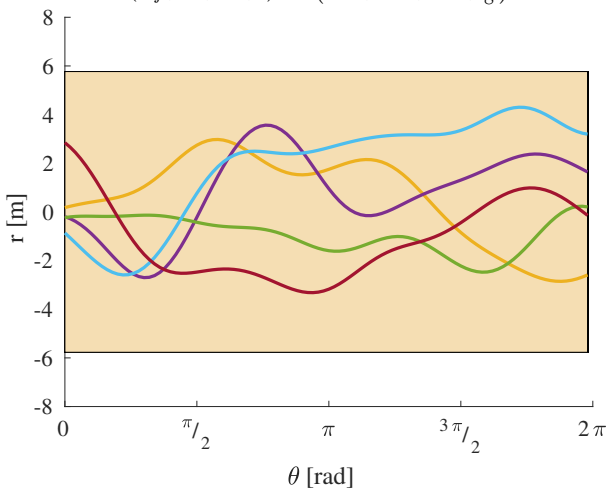
(b) $(\sigma_f, \sigma_b, \sigma_n, l) = \left(1\,\mathrm{m}, 0\,\mathrm{m}, 0\,\mathrm{m}, \frac{\pi}{4}\right)$.



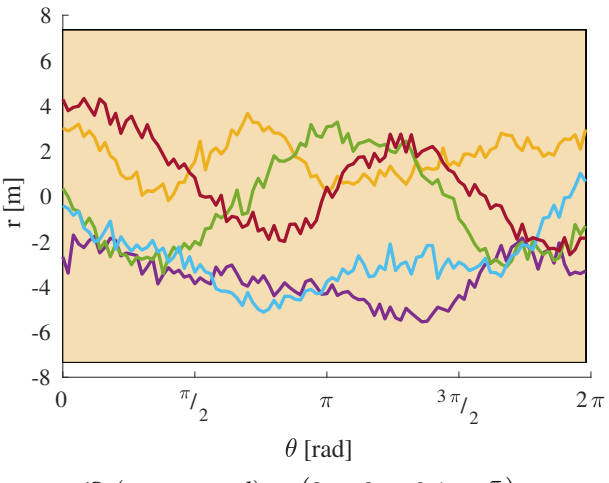
(c) $(\sigma_f, \sigma_b, \sigma_n, l) = \left(2\,\mathrm{m}, 0\,\mathrm{m}, 0\,\mathrm{m}, \frac{\pi}{8}\right)$.



(d) $(\sigma_f, \sigma_b, \sigma_n, l) = \left(2\,\mathrm{m}, 2\,\mathrm{m}, 0\,\mathrm{m}, \frac{\pi}{4}\right)$.



(e) $(\sigma_f, \sigma_b, \sigma_n, l) = \left(2\,\mathrm{m}, 1\,\mathrm{m}, 0\,\mathrm{m}, \frac{\pi}{4}\right)$.



(f) $(\sigma_f, \sigma_b, \sigma_n, l) = \left(2\,\mathrm{m}, 2\,\mathrm{m}, 0.1\,\mathrm{m}, \frac{\pi}{4}\right)$.

Figure 5.2: Five functions drawn from $\mathcal{GP}(0, k_e)$ for different hyperparameter values. The interval $[0, 2\pi)$ was regularly discretized using 100 angles. The shadowed region corresponds to a 99% confidence interval for each function value.

Figures 5.2a to 5.2e show only smooth functions. This is because the covariance k_e is a positive and smooth function as long as $\sigma_n = 0$, which implies that the values of the drawn functions at different points are always correlated in a continuous way. A smooth radius function at every point, may be undesirable in some applications. For example, if the contour of an object has sharp edges, as the extent of many maritime vessels do, then so will its radius function. Therefore, in order to model sharp edges and abnormalities at particular function values, the variance of the noise σ_n can be set to a positive value. By doing so, white random noise $w(\theta)$ is added to the drawn functions according to $w(\theta) \sim \mathcal{GP}(0, \sigma_n^2 \delta(\theta, \theta'))$. The resulting effect can be observed by comparing Figure 5.2f to the other subfigures in Figure 5.2.

The hyperparameter σ_n plays also another important role from a numerical point of view. If $\sigma_n = 0$, then the covariance matrix associated to an index set $\mathbb{I}$, $\mathbf{K}_e(\mathbb{I}, \mathbb{I})$, may be singular or ill-conditioned because its columns are too similar to each other. This is a major problem for GP regression, which requires the inverse of $\mathbf{K}_e(\mathbb{I}, \mathbb{I})$, as we will see Section 5.2. By setting the variance of the noise σ_n to a positive value, the matrix $\mathbf{K}_e(\mathbb{I}, \mathbb{I})$ is regularized by adding to it the matrix $\sigma_n^2 \mathbf{I}$.

Singular or ill-conditioned covariance matrices are a recurrent issue in GP models, and addition of a positive multiple of the identity matrix, also known as a "nugget", is a well-used regularization technique ([28]).

5.1.2 Gaussian processes for closed curves

As shown in the examples in Section 5.1.1, the covariance function k_e (5.3) allows us to model a wide variety of different radius functions r. However, the cartesian curves associated to these functions may or may not be closed, depending whether $r(0) = r(2\pi)$ or not.

As mentioned in the introduction to this chapter, we are mainly interested in using GPes to model radius functions that define closed curves, i.e., that are 2π-periodic. In addition, it is desirable that these GPes have the same inherent versatility as the GPes that use the covariance function k_e (see Section 5.1.1). Both objectives can be achieved by replacing the pure exponential term in k_e:

$$\tilde{k}_e(\theta, \theta') = \sigma_f^2 e^{-\frac{(\theta-\theta')^2}{2l^2}}, \tag{5.4}$$

by the term

$$\tilde{k}_{2\pi}(\theta, \theta') = \sigma_f^2 e^{-\frac{2}{l^2}\sin^2\left(\frac{\theta-\theta'}{2}\right)}. \tag{5.5}$$

This yields the covariance function

$$k_{2\pi}(\theta, \theta') = \sigma_f^2 e^{-\frac{2}{l^2}\sin^2\left(\frac{\theta-\theta'}{2}\right)} + \sigma_b^2 + \sigma_n^2 \delta(\theta, \theta'). \tag{5.6}$$

The same observations done in Section 5.1.1 about the effects that the hyperparameters σ_f, σ_b, σ_n and l have on the functions modeled by $\mathcal{GP}(0, k_e)$, also apply for the GPes of the form $\mathcal{GP}(0, k_{2\pi})$. This is a consequence of the limit

$$\lim_{\theta' \to \theta} \frac{\ln(\tilde{k}_{2\pi}(\theta, \theta'))}{\ln(\tilde{k}_e(\theta, \theta'))} = \lim_{\theta' \to \theta} \frac{-\frac{2}{l^2}\sin^2\left(\frac{\theta-\theta'}{2}\right)}{-\frac{(\theta-\theta')^2}{2l^2}} = \lim_{\theta' \to \theta} \left(\frac{\sin\left(\frac{\theta-\theta'}{2}\right)}{\frac{\theta-\theta'}{2}} \right)^2 = 1, \tag{5.7}$$

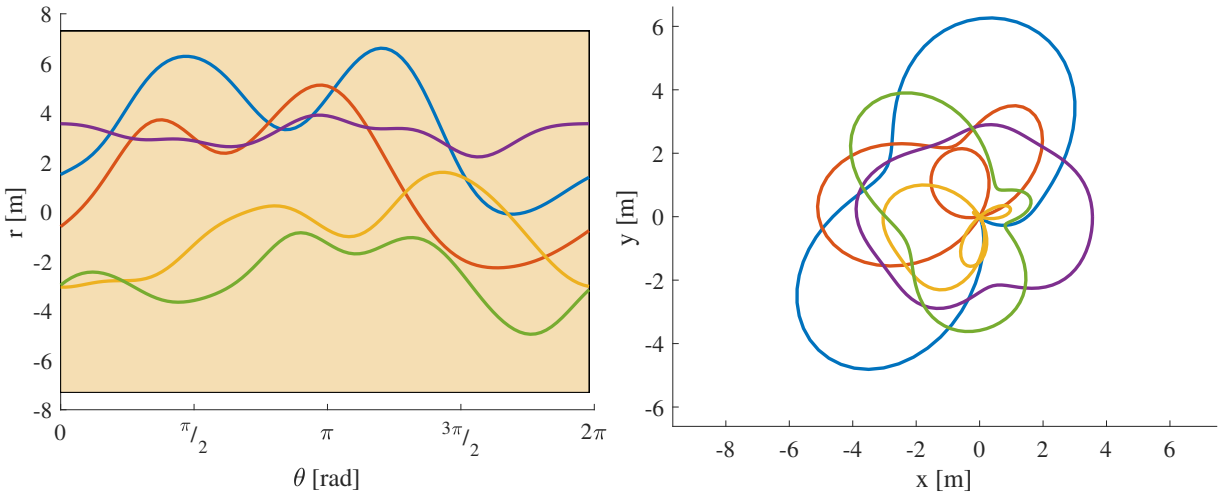
which implies that $k_{2\pi}(\theta, \theta') \to k_e(\theta, \theta')$ exponentially when $\theta' \to \theta$. Therefore, the radius functions modeled by GPes that use the covariance function $k_{2\pi}$ are locally similar to radius functions modeled by GPes that use the covariance function k_e for the same set of hyperparameter values.

Finally, we prove that $k_{2\pi}$ indeed models 2π-periodic functions: As a consequence of the definition of $k_{2\pi}$ (5.6), we obtain that the correlation of the function values at the angles θ and $\theta + 2\pi$ is

$$\rho_{2\pi}(\theta, \theta + 2\pi) = \frac{k_{2\pi}(\theta, \theta + 2\pi)}{\sqrt{k_{2\pi}(\theta, \theta)}\sqrt{k_{2\pi}(\theta + 2\pi, \theta + 2\pi)}} = \frac{\sigma_f^2 + \sigma_b^2}{\sigma_f^2 + \sigma_b^2 + \sigma_n^2}. \tag{5.8}$$

Therefore, if there is no white random noise term, i.e. $\sigma_n = 0$, this correlation is equal to 1, which implies that the function values at θ and $\theta + 2\pi$ have to be the same. In other words, if no white random noise is added, $\mathcal{GP}(0, k_{2\pi})$ models radius functions that are 2π-periodic.

On the other hand, if there is a non-zero white random noise term in $k_{2\pi}$, i.e. $\sigma_n > 0$, then these correlations are less than 1, and the modeled radius functions do not necessarily satisfy the above-mentioned properties, which is as expected due to the nature of white random noise.



(a) Radius functions. **(b)** Closed cartesian curves.

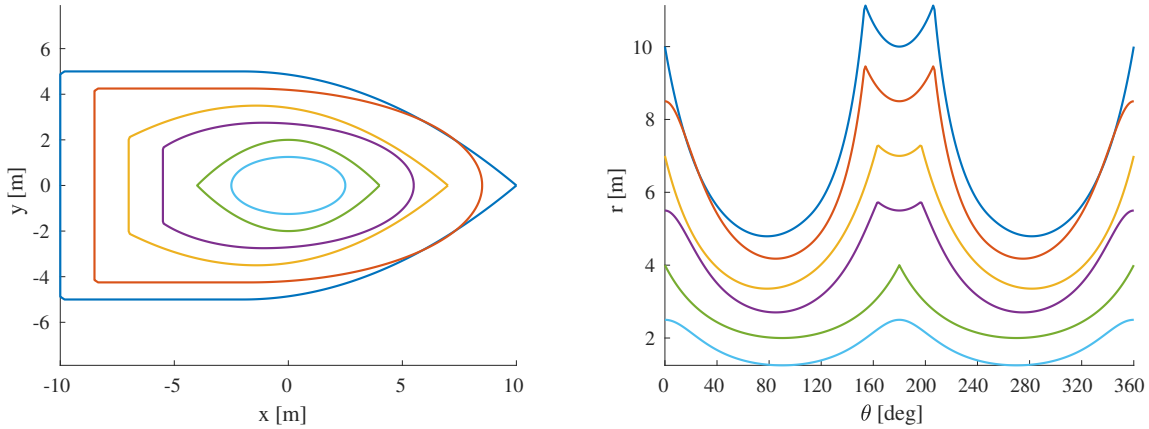
Figure 5.3: Five radius functions drawn from $\mathcal{GP}(0, k_{2\pi})$ with $(\sigma_f, \sigma_b, \sigma_n, l) = (2\,\text{m}, 2\,\text{m}, 0\,\text{m}, \frac{\pi}{4})$, and their respective closed cartesian curves. The interval $[0, 2\pi)$ was regularly discretized using 100 angles. The shadowed region corresponds to a 99% confidence interval for each function value.

Figure 5.3 illustrates the radius functions modeled by the GPes of the form $\mathcal{GP}(0, k_{2\pi})$ for $\sigma_n = 0$, as well as their associated cartesian curves. As done in the examples shown in Figure 5.2, the interval $[0, 2\pi)$ was regularly discretized using 100 angles, and five radius functions were drawn from the resulting multivariate Gaussian distribution. The hyperparameters used were $(\sigma_f, \sigma_b, \sigma_n, l) = \left(2\,\text{m}, 2\,\text{m}, 0\,\text{m}, \frac{\pi}{4}\right)$. As shown in Figure 5.3b, the associated cartesian curves are indeed closed, as well as smooth since $\sigma_n = 0$. In addition, they have a wide range of shapes.

5.1.3 Gaussian processes for closed and axis-symmetric curves

As shown in Section 5.1.2, GPes of the form $\mathcal{GP}(0, k_{2\pi})$ can model a wide variety of radius functions, which define closed cartesian curves due to their inherent 2π-periodicity.

As discussed in the introduction to this chapter, the extents of interest in this thesis resemble hulls of maritime vessels in the horizontal plane. Most ship hulls are symmetric about their longitudinal axes, or they can be reasonably assumed so. Figure 5.4 shows examples of boat hulls and they respective radius functions about their center. As we can observe, the axis symmetry of the cartesian curves (Figure 5.4a) is equivalent to the radius functions being even functions, in addition to 2π-periodic (Figure 5.4b).



(a) Cartesian curves of boat hulls.

(b) Radius functions of boat hulls.

Figure 5.4: Boat hull examples.

If the axial symmetry of the extents is modeled into the used GPes, this may drastically improve the estimation of the extent, because measurements related directly to some function values can be used to estimate other function values due to symmetry. Therefore, it is desirable to further modify the covariance function $k_{2\pi}$ in order to model 2π-periodic and even radius functions.

This is achieved by replacing the pure exponential term $\tilde{k}_{2\pi}$ in $k_{2\pi}$ (5.5) by the term

$$\tilde{k}_{2\pi,a}(\theta, \theta') = \sigma_f^2 e^{-\frac{1}{2l^2}(|\operatorname{ssa}(\theta)| - |\operatorname{ssa}(\theta')|)^2}, \tag{5.9}$$

where $\operatorname{ssa}(\cdot)$ is the smallest signed angle function, which is defined as

$$\operatorname{ssa}(\theta) = \pi - [(\pi - \theta) \pmod{2\pi}], \tag{5.10}$$

i.e. $\operatorname{ssa}(\theta)$ is the only angle in $(-\pi, \pi]$ such that $\operatorname{ssa}(\theta) \equiv \theta \pmod{2\pi}$.

The absolute value of the smallest signed angle function, $|\operatorname{ssa}(\cdot)|$ is illustrated in Figure 5.5. As we can observe in this figure, this function is constructed in such way that

$$|\operatorname{ssa}(\theta)| = |\operatorname{ssa}(-\theta)| \tag{5.11a}$$
$$|\operatorname{ssa}(\theta)| = |\operatorname{ssa}(\pi - \theta)|. \tag{5.11b}$$

By replacing $\tilde{k}_{2\pi}$ by $\tilde{k}_{2\pi,a}$ in $k_{2\pi}$ (5.6), we obtain the covariance function

$$k_{2\pi,a}(\theta,\theta') = \sigma_f^2 e^{-\frac{1}{2l^2}(|\operatorname{ssa}(\theta)|-|\operatorname{ssa}(\theta')|)^2} + \sigma_b^2 + \sigma_n^2\delta(\theta,\theta'). \tag{5.12}$$

As is the case with $\mathcal{GP}(0, k_{2\pi})$, the hyperparameters σ_f, σ_b, σ_n and l have the same effects on the functions modeled by $\mathcal{GP}(0, k_{2\pi,a})$, as they do in the GPes of the form $\mathcal{GP}(0, k_e)$. This is because $k_{2\pi,a}(\theta,\theta') \to k_e(\theta,\theta')$ exponentially when $\theta' \to \theta$, as shown by the following limit

$$\lim_{\theta'\to\theta} \frac{\ln(\tilde{k}_{2\pi,a}(\theta,\theta'))}{\ln(\tilde{k}_e(\theta,\theta'))} = \lim_{\theta'\to\theta} \frac{-\frac{2}{l^2}\left(|\operatorname{ssa}(\theta)| - |\operatorname{ssa}(\theta')|\right)^2}{-\frac{(\theta-\theta')^2}{2l^2}} \tag{5.13a}$$

$$= \lim_{\theta'\to\theta} \left(\frac{|\operatorname{ssa}(\theta)| - |\operatorname{ssa}(\theta')|}{\theta-\theta'}\right)^2 = 1\,, \tag{5.13b}$$

where the last limit follows from $|\operatorname{ssa}(\cdot)|$ being a saw-tooth function as shown in Figure 5.5. In particular, we have that the radius functions modeled by GPes that use the covariance function $k_{2\pi,a}$ are locally similar to radius functions modeled by GPes that use the covariance function k_e for the same set of hyperparameter values.

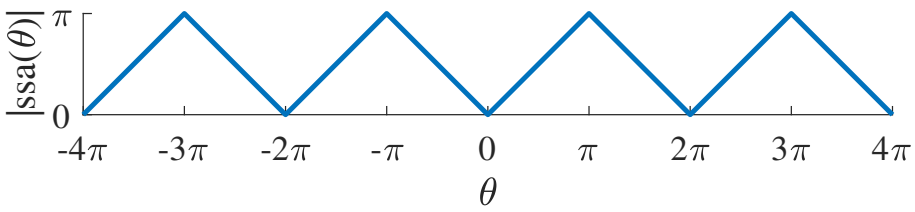


Figure 5.5: $|\operatorname{ssa}(\theta)|$ for $\theta \in [-4\pi, 4\pi]$.

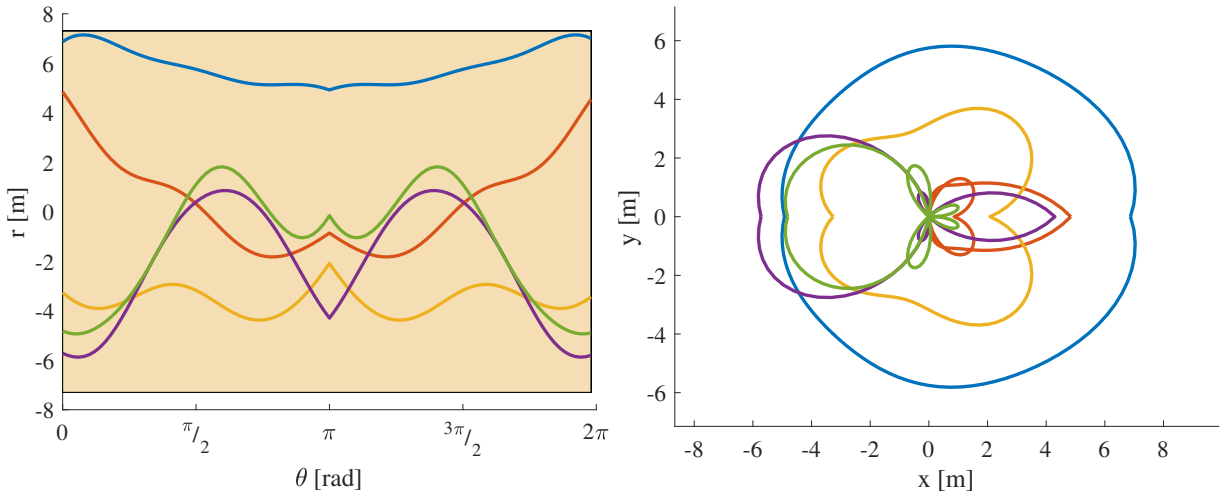
Finally, we prove that $k_{2\pi,a}$ indeed models 2π-periodic function and even functions: As a consequence of the definition of $k_{2\pi,a}$ (5.12) and the properties in (5.11), we obtain the following correlations between the function values as θ, $\theta + 2\pi$ and $-\theta$:

$$\rho_{2\pi,a}(\theta,\theta+2\pi) = \frac{k_{2\pi,a}(\theta,\theta+2\pi)}{\sqrt{k_{2\pi,a}(\theta,\theta)}\sqrt{k_{2\pi,a}(\theta+2\pi,\theta+2\pi)}} = \frac{\sigma_f^2+\sigma_b^2}{\sigma_f^2+\sigma_b^2+\sigma_n^2} \tag{5.14a}$$

$$\rho_{2\pi,a}(\theta,-\theta) = \frac{k_{2\pi,a}(\theta,-\theta)}{\sqrt{k_{2\pi,a}(\theta,\theta)}\sqrt{k_{2\pi,a}(-\theta,-\theta)}} = \frac{\sigma_f^2+\sigma_b^2}{\sigma_f^2+\sigma_b^2+\sigma_n^2}. \tag{5.14b}$$

Therefore, if there is no white random noise term, i.e. $\sigma_n = 0$, all these correlations are equal to 1, and $\mathcal{GP}(0, k_{2\pi,a})$ models radius functions that are 2π-periodic and even, as desired. On the other hand, if there is a non-zero white random noise term in $k_{2\pi,a}$, then the modeled radius functions do not necessarily satisfy the above-mentioned properties, since the GP allows for uncorrelated function value variations at each point.

Figure 5.6 illustrates the radius functions modeled by the GP class $\mathcal{GP}(0, k_{2\pi,a})$ for $\sigma_n = 0$, as well as their associated cartesian curves. The discretization of the interval $[0, 2\pi)$ and the chosen hyperparameter values are the same as in the example of Figure 5.3. As we can observe, the associated cartesian curves are indeed closed and symmetric about their longitudinal axis.



(a) Radius functions. **(b)** Cartesian curves.

Figure 5.6: Five radius functions drawn from $\mathcal{GP}(0, k_{2\pi,a})$ with $(\sigma_f, \sigma_b, \sigma_n, l) = \left(2\,\mathrm{m}, 2\,\mathrm{m}, 0\,\mathrm{m}, \frac{\pi}{4}\right)$, and their respective closed and axis-symmetric cartesian curves. The interval $[0, 2\pi)$ was regularly discretized using 100 test angles. The shadowed region corresponds to a 99% confidence interval for each function value.

The covariance functions k_e and $k_{2\pi}$ can be found in the literature (see for example [30] and [37]). However, the covariance function $k_{2\pi,a}$ is not present in the literature to the best of the author's knowledge, and this covariance function was explicitly developed for this study in order to model axial symmetry. In [20], a covariance function for modelling axial symmetry is present. However, the maximum covariance values of this function are in some cases not achieved at identical angles, but at angles that are slightly shifted. Therefore, this covariance function has not been chosen.

5.2 Gaussian process regression

GP regression is a method for estimating the values of an unknown function that is modeled as a GP based on noisy measurements of some of its function values. In this section, we present the theoretical background and important equations for this regression method, as well as an example that involves the covariance functions defined in Section 5.1.

Let $f(x) \sim \mathcal{GP}(m(x), k(x, x'))$ be a GP, where $x, x' \in \mathbb{I}$ and $\mathbb{I} \subset \mathbb{R}$ is an interval, and let $\mathbf{x}_t = [x_{t,1}, x_{t,2}, \cdots, x_{t,N_t}]^T = [x_t]_{t=1}^{N_t} \in \mathbb{I}^{N_t}$ be a vector of different points in $\mathbb{I}$, which are called test points. The test points are points at which the function values are of special interest. Usually, the test points correspond to a discretization of the interval $\mathbb{I}$.

The measurements $\mathbf{y} = [y_1, y_2, \cdots, y_{N_i}] = [y_n]_{n=1}^{N_i} \in \mathbb{R}^{N_i}$ of the function values at the points $\mathbf{x}_i = [x_{i,1}, x_{i,2}, \cdots, x_{i,N_i}] = [x_{i,n}]_{n=1}^{N_i} \in \mathbb{I}^{N_i}$ are modeled as

$$y_n = f(x_{i,n}) + w_n\,, \tag{5.15}$$

where $w_n \sim \mathcal{N}(0, \sigma_w^2)$ is the measurement noise and $\sigma_w > 0$ is the measurement noise strength.

Furthermore, the noise at each measurement is assumed independent of each other, as well as independent of the function values.

The points at which the function values are measured, $\mathbf{x}_i = [x_{i,n}]_{n=1}^{N_i}$ are called input points. The denominations "test" and "input" points have their origin in the Machine Learning field (see [30]).

The GP model definition (Definition 5.1.1) and the independence assumption on the measurement noise yields that

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_t \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{M}(\mathbf{x}_i) \\ \mathbf{M}(\mathbf{x}_t) \end{bmatrix}, \begin{bmatrix} \mathbf{K}(\mathbf{x}_i, \mathbf{x}_i) + \sigma_w^2 \mathbf{I}_{N_i} & \mathbf{K}(\mathbf{x}_i, \mathbf{x}_t) \\ \mathbf{K}(\mathbf{x}_t, \mathbf{x}_i) & \mathbf{K}(\mathbf{x}_t, \mathbf{x}_t) \end{bmatrix} \right), \tag{5.16}$$

where $\mathbf{f}_i = [f(x_{i,1}), f(x_{i,2}), \cdots f(x_{i,N_i})]^T \in \mathbb{R}^{N_i}$ and $\mathbf{f}_t = [f(x_{t,1}), f(x_{t,2}), \cdots f(x_{t,N_t})]^T \in \mathbb{R}^{N_t}$ are the function values at the input and test points, respectively.

By using Theorem A.1 from Appendix A, we conclude that the conditional distribution of $\mathbf{f}_t|\mathbf{y}$ is normally distributed with mean vector and covariance matrix given by

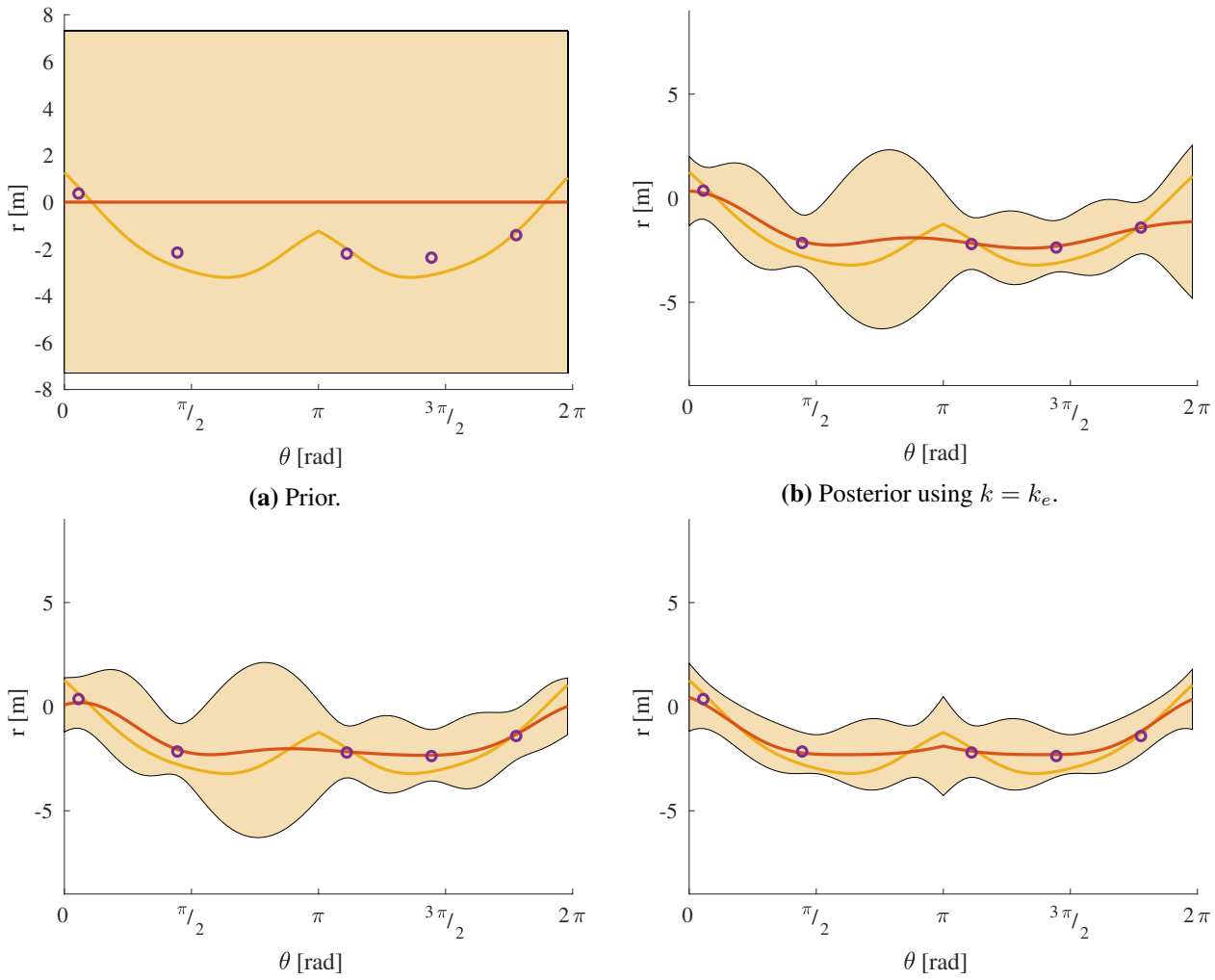
$$\mathbb{E}[\mathbf{f}_t|\mathbf{y}] = \mathbf{M}(\mathbf{x}_t) + \mathbf{K}(\mathbf{x}_t, \mathbf{x}_i)[\mathbf{K}(\mathbf{x}_i, \mathbf{x}_i) + \sigma_w^2 \mathbf{I}_{N_i}]^{-1} (\mathbf{y} - \mathbf{M}(\mathbf{x}_i)) \tag{5.17a}$$

$$\mathit{Cov}[\mathbf{f}_t|\mathbf{y}] = \mathbf{K}(\mathbf{x}_t, \mathbf{x}_t) - \mathbf{K}(\mathbf{x}_t, \mathbf{x}_i)[\mathbf{K}(\mathbf{x}_i, \mathbf{x}_i) + \sigma_w^2 \mathbf{I}_{N_i}]^{-1} \mathbf{K}[\mathbf{x}_i, \mathbf{x}_t] \tag{5.17b}$$

Equations (5.17a) to (5.17b) summarize GP regression: The prior distribution of the function values at the test points is $\mathbf{f}_t \sim \mathcal{N}(\mathbf{M}(\mathbf{x}_t), \mathbf{K}(\mathbf{x}_t, \mathbf{x}_t))$ due to the GP model assumption, and given the measurements $\mathbf{y}$, the posterior distribution of these function values, $\mathbf{f}_t|\mathbf{y}$, is normally distributed with mean vector and covariance matrix given by Equation (5.17a) and Equation (5.17b), respectively.

Figure 5.7 shows an GP regression example. As done in previous examples (see Section 5.1), the test points correspond to a regular discretization of the interval $[0, 2\pi)$ of 100 points. The true function to be estimated is drawn from $\mathcal{GP}(0, k_{2\pi,a})$ with hyperparameters $(\sigma_f, \sigma_b, \sigma_n, l) = \left(2\,\mathrm{m}, 2\,\mathrm{m}, 0, \frac{\pi}{4}\right)$, and 5 measurements are taken with $\sigma_w = 0.5\,\mathrm{m}$ at 5 different input points on the interval $[0, 2\pi)$. GP regression is performed using $\mathcal{GP}(0, k_e)$, $\mathcal{GP}(0, k_{2\pi})$ and $\mathcal{GP}(0, k_{2\pi,a})$.

Figure 5.7a shows the prior distribution of $\mathbf{f}_t$: The mean vector (red) and a 99% confidence interval for each function value (shadowed region) is the same for each considered GP. Figures 5.7b to 5.7d show the GP regression results using $\mathcal{GP}(0, k_e)$, $\mathcal{GP}(0, k_{2\pi})$ and $\mathcal{GP}(0, k_{2\pi,a})$, respectively. In each case, the mean vector of the posterior distribution $\mathbf{f}_t|\mathbf{y}$ is close to the true function values, specially for points that lie near the input points. Moreover, the true function lies inside the 99% confidence region for the estimated function values, which is considerably narrower than the one for the prior distribution. The results using the covariance function $k_{2\pi}$ are better than the ones obtained using k_e since $\mathcal{GP}(0, k_{2\pi})$ correctly assumes that the true function is 2π-periodic, and uses the measurements to estimate the function values at test points that are 2π units away of the input points. This explains why the estimate at test points close to 0 and 2π are considerably better than the estimates obtained using k_e, and the confidence region is narrower. By a similar argument, we can explain why the estimates using $k_{2\pi,a}$ are even better than the ones obtained using $k_{2\pi}$. This is because a measurement at an input point θ_i is in



(a) Prior.

(b) Posterior using $k = k_e$.

(c) Posterior using $k = k_{2\pi}$.

(d) Posterior using $k = k_{2\pi,a}$.

Figure 5.7: GP regression: True function (yellow), measurements (purple), mean of $\mathbf{f}_t$ (red) and a 99% confidence region for $\mathbf{f}_t$ (shadowed region). Hyperparameter values $(\sigma_f, \sigma_b, \sigma_n, l) = \left(2\,\mathrm{m}, 2\,\mathrm{m}, 0, \frac{\pi}{4}\right)$.

practice also a measurement at the point $\pi - \theta$ since the GP model correctly assumes that the unknown function is 2π-periodic and even.

In particular, this example shows that the incorporation of a priori information about the unknown function into the GP model may give superior estimation results.

Chapter 6

State-space models

In Chapter 4, the PMBM filter for MEOT was derived for general stochastic models for both the dynamics of the object's state (see **A.4**) and the object-generated measurements (see **B.4**). In this chapter, we present the particular models that are used in the simulations (see Chapter 7).

Two different state-space models are presented. The first one models the object extent using GPes, and its measurement model simulates lidar measurements, while the second model approximates the object extent by an ellipsis and simulates radar measurements.

The GP and lidar state-space model is the central state-space model of this thesis and is the one used in [37] and [20], while the second model, which is called the gamma Gaussian inverse Wishart (GGIW) is of secondary importance, and is briefly presented. The GGIW model is by default implemented in the PMBM code by Yuxuan Xia in [41]. This code was modified to work with the GP and lidar model. Therefore, the GGIW model is used in Chapter 7 as a reference for what can be achieved with the PMBM filter for MEOT.

In both the GP lidar model and the GGIW model, the state vector $\mathbf{x}$ of any object consist of a set of kinematic states $\mathbf{x}_c$ that describe the movement of the object, and of a set of extent states $\mathbf{x}_e$ that describe the geometry of the object's extent. In symbols, $\mathbf{x} = [\mathbf{x}_c^T \quad \mathbf{x}_e^T]^T$.

6.1 State-transition model for kinematic states

6.1.1 Reference frames and kinematic states

The position and velocity of all objects is expressed with respect to a fixed North-East-Down (NED) reference frame (see [13]), which is referred to as the world frame. NED frames are strictly speaking local and non-inertial reference frames. However, since the objects move during a relatively short time-span, and they do not move relatively far away, nor fast in a planetary scale, the world frame can be assumed to be inertial for our purposes.

For each object, there is a fixed point called the center of movement, whose position and velocity in the world frame is representative for the whole object. Since objects are assumed to be rigid bodies, this convention is well-defined.

Furthermore, since the MEOT problem is restricted to the horizontal plane, only the North and East coordinates are relevant. Therefore, the position of the object is $\mathbf{r}_c = [N_c\,, E_c]^T$, where N_c and E_c are the North and East coordinates of the center of movement, respectively. Analogously, the velocity of the object is $\mathbf{v}_c = [v_{N_c}\,, v_{E_c}]^T$, where v_{N_c} and v_{E_c} are the velocities of the object in the North and East directions, respectively.

Since the world frame is inertial, we have the relation $\mathbf{v}_c = \dot{\mathbf{r}}_c$, i.e.,

$$v_{N_c} = \dot{N}_c \tag{6.1a}$$

$$v_{E_c} = \dot{E}_c. \tag{6.1b}$$

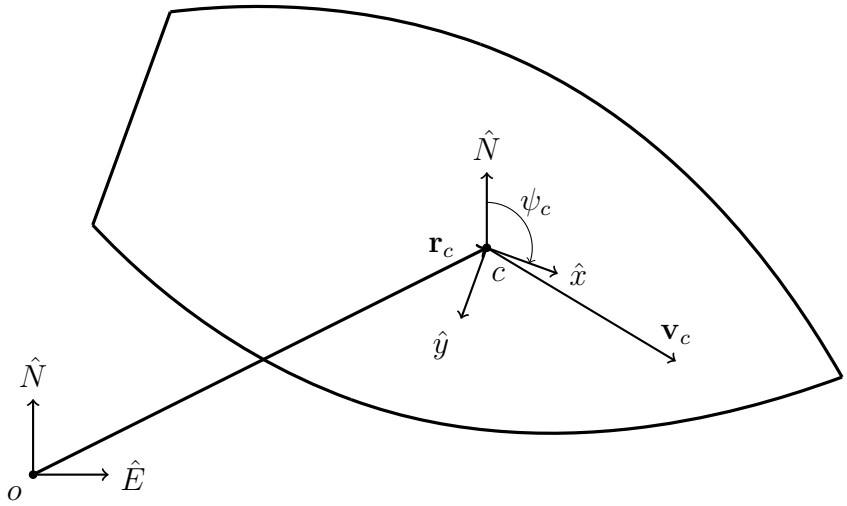


Figure 6.1: A maritime vessel with position $\mathbf{r}_c = [N_c,\, E_c]^T$, velocity $\mathbf{v}_c = [v_{N_c},\, v_{E_c}]^T$ and heading ψ_c. The world frame has origin o and axes $\hat{N}$ (North) and $\hat{E}$ (East), while the body frame has origin c and axes $\hat{x}$ and $\hat{y}$.

Due to possible sideslip of the maritime vessel, the velocity vector $\mathbf{v}_c$ does not necessarily give the orientation of the object's extent. Therefore, the heading angle ψ_c, which gives the orientation of the extent with respect to the world frame, is also a kinematic quantity of interest. Formally, the heading angle is defined as the rotation angle between the world frame and a local reference frame that is attached to the extent, which is known as the body frame. This moving frame has the center of movement as its origin, and its axes usually correspond to important directions, such as symmetry axes. Since the extent represents a maritime vessel in our case, the first body axis is chosen to give the stern to bow direction, while the second axis gives the port to starboard direction. The relation between the world and body frames, as well as the introduced kinematic states N_c, E_c, v_{N_c}, v_{E_c} and ψ_c, is illustrated in Figure 6.1.

Finally, we have the turn rate r_c, which is the derivative of the heading angle r_c, i.e., $r_c = \dot{\psi}_c$. This kinematic state describes the angular velocity of the extent.

Hence, the possible kinematic states in the models are N_c, E_c, v_{N_c}, v_{E_c}, ψ_c and r_c.

6.1.2 The constant velocity model

The constant velocity (CV) model is a well-used kinematic model that models the acceleration of an object as white Gaussian random noise. Therefore, the velocity of the object is more or less constant with variations that depend on the noise strength.

In the one-dimensional case, the state vector is $\mathbf{x}(t) = [x(t), v(t)]^T$, where $x(t)$ and $v(t)$ are the position and velocity of the object, respectively. Hence, the continuous state-space model is given by

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{G}w(t)\,, \tag{6.2}$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \tag{6.3a}$$

$$\mathbf{G} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{6.3b}$$

and $w(t)$ is white Gaussian random noise of strength σ_w, i.e.

$$w(t) \sim \mathcal{N}\left(0, \sigma_w^2 \delta(t-\tau)\right)\,, \tag{6.4}$$

where $\delta(\tau)$ is the Dirac-delta.

The solution of the differential equation in (6.2) is

$$\mathbf{x}(t) = e^{\mathbf{A}(t-t_0)}\mathbf{x}(t_0) + \int_{t_0}^{t} e^{\mathbf{A}(t-\tau)}\mathbf{G}w(\tau)\,d\tau. \tag{6.5}$$

If the solution is discretized with a constant time-step T, i.e. the solution is sampled at time points t_k given by $t_k = t_0 + Tk$, $k \in \mathbb{N} \cup \{0\}$, then the exact values at these time points are given by

$$\mathbf{x}(t_k) = e^{\mathbf{A}T}\mathbf{x}(t_{k-1}) + \int_{t_{k-1}}^{t_k} e^{\mathbf{A}(t_k-\tau)}\mathbf{G}w(\tau)\,d\tau. \tag{6.6}$$

By denoting $\mathbf{x}_k = \mathbf{x}(t_k)$ and

$$\mathbf{w}_k = \int_{t_{k-1}}^{t_k} e^{\mathbf{A}(t_k-\tau)}\mathbf{G}w(\tau)\,d\tau\,, \tag{6.7}$$

we can rewrite the exact discretization as

$$\mathbf{x}_k = e^{\mathbf{A}T}\mathbf{x}_{k-1} + \mathbf{w}_k. \tag{6.8}$$

Since $\mathbf{A}^n = 0$ for $n \geq 2$, it follows that

$$e^{\mathbf{A}T} = \mathbf{I} + \mathbf{A}T + \sum_{n=2}^{\infty} \frac{\mathbf{A}^n T^n}{n!} = \mathbf{I} + \mathbf{A}T = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}. \tag{6.9}$$

Moreover, $\mathbf{w}_k$ is a stochastic variables that is normally distributed with

$$\mathbb{E}[\mathbf{w}_k] = \int_{t_{k-1}}^{t_k} e^{\mathbf{A}(t_k-\tau)}\mathbf{G}\mathbb{E}[w(\tau)]\,d\tau = 0 \tag{6.10a}$$

$$\mathbb{V}[\mathbf{w}_k] = \int_{t_{k-1}}^{t_k} e^{\mathbf{A}(t_k-\tau)}\mathbf{G}\mathbb{E}[w(\tau)w(\tau)^T]\mathbf{G}^T e^{\mathbf{A}(t_k-\tau)^T}\,d\tau \tag{6.10b}$$

$$= \int_{t_{k-1}}^{t_k} \begin{bmatrix} 1 & t_k-\tau \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \sigma_w^2 \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ t_k-\tau & 1 \end{bmatrix} d\tau \tag{6.10c}$$

$$= \sigma_w^2 \int_{t_{k-1}}^{t_k} \begin{bmatrix} (t_k-\tau)^2 & t_k-\tau \\ t_k-\tau & 1 \end{bmatrix} d\tau = \sigma_w^2 \begin{bmatrix} \frac{T^3}{3} & \frac{T^2}{2} \\ \frac{T^2}{2} & T \end{bmatrix} \tag{6.10d}$$

Therefore, the discrete one-dimensional constant velocity model can be stated as

$$\mathbf{x}_k = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \mathbf{x}_{k-1} + \mathbf{w}_k\,, \qquad \mathbf{w}_k \sim \mathcal{N}\left(\mathbf{0}, \sigma_w^2 \begin{bmatrix} \frac{T^3}{3} & \frac{T^2}{2} \\ \frac{T^2}{2} & T \end{bmatrix}\right), \tag{6.11}$$

where the initial state $\mathbf{x}_0 = \mathbf{x}(t_0)$ is assumed to be distributed as $\mathcal{N}(\bar{\mathbf{x}}_0, \mathbf{P}_0)$ for some initial state estimate $\bar{\mathbf{x}}_0$ and a covariance matrix $\mathbf{P}_0$.

Note that the units of σ_w in

$$\mathcal{N}\left(\mathbf{0}, \sigma_w^2 \begin{bmatrix} \frac{T^3}{3} & \frac{T^2}{2} \\ \frac{T^2}{2} & T \end{bmatrix}\right) \tag{6.12}$$

have to be $\mathrm{m\,s}^{3/2}$, which is an unit that is hard to grasp. Therefore, we rescale the covariance matrix to be

$$\mathbf{w}_k \sim \mathcal{N}\left(\mathbf{0}, \sigma_w^2 \begin{bmatrix} \frac{T^2}{3} & \frac{T}{2} \\ \frac{T}{2} & 1 \end{bmatrix}\right), \tag{6.13}$$

where the units of σ_w are the ones of velocity.

Initial state distribution using finite differences

Let us assume that as part of the detection process, measurements of the object position are available for the time points t_{-1} and t_0. Let us denote these estimates by $\bar{x}_0$ and $\bar{x}_{-1}$, respectively, and assume that the position measurement noise has zero mean and variance σ_x^2, and that measurements at different time points are uncorrelated.

An estimate of the velocity at t_0, $\bar{v}_0$, can be obtained by using the finite difference

$$\bar{v}_0 = \frac{\bar{x}_0 - \bar{x}_{-1}}{T} \tag{6.14}$$

If the time step T is small enough such that the finite difference is a good approximation of the velocity at t_0, it follows from this assumption and the assumptions on the measurement

noise that

$$\mathbb{E}[x_0] = \bar{x}_0 \tag{6.15a}$$

$$\mathbb{E}[v_0] = \frac{\bar{x}_0 - \bar{x}_{-1}}{T} = \bar{v}_0 \tag{6.15b}$$

$$\mathbb{E}[(x_0 - \bar{x}_0)^2] = \sigma_x^2 \tag{6.15c}$$

$$\mathbb{E}[(x_0 - \bar{x}_0)(v_0 - \bar{v}_0)] = \frac{1}{T}\mathbb{E}[(x_0 - \bar{x}_0)^2] - \frac{1}{T}\mathbb{E}[(x_0 - \bar{x}_0)(x_{-1}) - \bar{x}_{-1})] = \frac{\sigma_x^2}{T} \tag{6.15d}$$

$$\mathbb{E}[(v_0 - \bar{v}_0)^2] = \frac{1}{T^2}\mathbb{E}[(x_0 - \bar{x}_0)^2] + \frac{1}{T^2}\mathbb{E}[(x_{-1} - \bar{x}_{-1})^2] = \frac{2\sigma_x^2}{T^2}. \tag{6.15e}$$

The expected values, covariances and variances of (6.15) motivate the following model for the initial state distribution

$$\mathbf{x}_0 \sim \mathcal{N}\left(\begin{bmatrix}\bar{x}_0 \\ \bar{v}_0\end{bmatrix}, \sigma_x^2 \begin{bmatrix}1 & \frac{1}{T} \\ \frac{1}{T} & \frac{2}{T^2}\end{bmatrix}\right). \tag{6.16}$$

6.1.3 CV2 model without sideslip

The state-transition model for the kinematic states is modeled as 2 decoupled constant velocity (CV) models (see Section 6.1.2) for the position in the North and East directions. Hence, the kinematic state vector is

$$\mathbf{x}_c = \begin{bmatrix} N_c & E_c & v_{N_c} & v_{E_c} \end{bmatrix}^T, \tag{6.17}$$

and the kinematic state-transition model is

$$\mathbf{x}_{c,k} = \mathbf{F}_c \mathbf{x}_{c,k-1} + \mathbf{w}_{c,k}, \qquad \mathbf{w}_{c,k} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_c), \tag{6.18}$$

where

$$\mathbf{F}_c = \begin{bmatrix}1 & T \\ 0 & 1\end{bmatrix} \otimes \mathbf{I}_2 \tag{6.19a}$$

$$\mathbf{Q}_c = \begin{bmatrix}\frac{T^2}{3} & \frac{T}{2} \\ \frac{T}{2} & 1\end{bmatrix} \otimes \begin{bmatrix}\sigma_N^2 & 0 \\ 0 & \sigma_E^2\end{bmatrix}, \tag{6.19b}$$

where σ_N and σ_E are the respective random noise strength for each coordinate direction.

Furthermore, we use (6.16) to model the distribution of the initial kinematic-state vector

$$\mathbf{x}_{c,0} \sim \mathcal{N}\left(\bar{\mathbf{x}}_{c,0}, \begin{bmatrix}1 & \frac{1}{T} \\ \frac{1}{T} & \frac{2}{T^2}\end{bmatrix} \otimes \begin{bmatrix}\sigma_{N_0}^2 & 0 \\ 0 & \sigma_{E_0}^2\end{bmatrix}\right), \tag{6.20}$$

where σ_{N_0} and σ_{E_0} are the respective standard deviations of the initial position estimate in each coordinate direction, $\bar{x}_{c,0}$ is an initial state estimate.

Regarding the heading angle ψ_c, we assume that the sideslip experienced by the maritime vessel is neglectable. Hence, the heading angle ψ_c corresponds to the direction of linear velocity vector $[v_{N_c}, v_{E_c}]^T$, i.e.,

$$\psi_c = \operatorname{atan2}(v_{N_c}, v_{E_c}) \pmod{2\pi}, \tag{6.21}$$

where $\mathrm{atan2}(\cdot,\cdot)$ is the four quadrant arctangent.

To neglect any sideslip is a reasonable assumption as long as no strong currents or winds are present, and as long as the turn rate of the maritime vessel is low compared to its velocity. If this is not the case, it is recommended to have a decoupled CV model for the heading in addition to the 2 decoupled CV models for the position.

6.2 Gaussian process and lidar model

6.2.1 Gaussian process extent model

The objective of the extent model is to give a general as possible parametrization of the object's extent. Since the extent resembles the hull of a maritime vessel in the horizontal plane, it is reasonable to assume that the object's extent is a star convex set about the center of movement.[1] Under this assumption, the object's extent is univocally determined by a polar parametrization in the body frame of the extent's boundary, as illustrated in Figure 6.2.

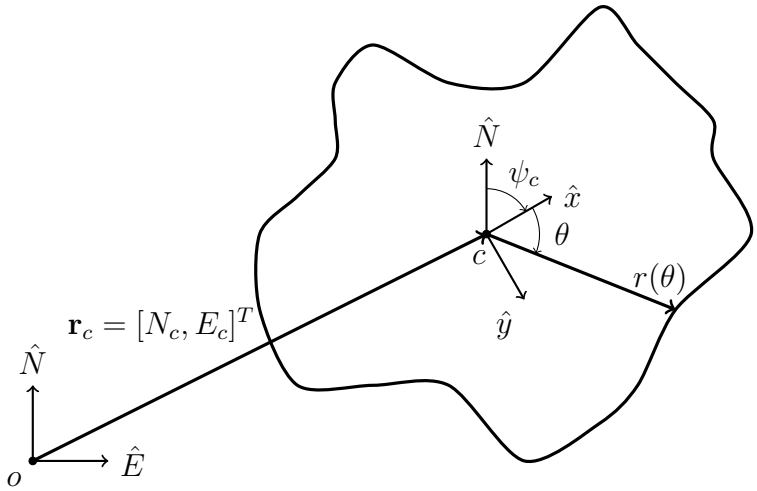


Figure 6.2: Polar parametrization in the body frame of the extent's boundary. The radius function $r(\theta)$ univocally determines the extent.

Let $r : [0, 2\pi] \to \mathbb{R}$ be the unknown radius function of the polar parametrization of the extent's boundary in the body frame. Then, the problem of determining the object's extent is equivalent to finding the radius function r. For this purpose, r is modeled as a GP of the form $\mathcal{G}(0, k_{2\pi,a})$ (see Section 5.1.3).

In practice, it is unnecessary -if not impossible- to obtain an analytical expression for $r(\theta)$ since a discretization of this function is sufficient to describe the extent. Therefore, let $\boldsymbol{\theta}_t = [\theta_{t,n}]_{n=1}^{N_t}$ be a fine enough fixed discretization of the interval $[0, 2\pi)$, i.e.,

$$0 = \theta_{t,1} < \theta_{t,2} < \cdots < \theta_{t,N_t} < 2\pi.$$

[1] A set $S \subset \mathbb{R}^n$ is star convex about a point $\mathbf{p} \in S$ if and only if the segment $[\mathbf{p}, \mathbf{q}] = \{\lambda\mathbf{p} + (1-\lambda)\mathbf{q} : \lambda \in [0,1]\}$ is contained in S for all $\mathbf{q} \in S$.

Then the extent is described by the values of the radius function r at these angles, and the extent states are defined as

$$\mathbf{x}_e = \begin{bmatrix} r(\theta_{t,1}) & r(\theta_{t,2}) & \cdots & r(\theta_{t,N_t}) \end{bmatrix}^T . \tag{6.22}$$

In particular, since we use a CV2 model with no sideslip to model the kinematics of the object (see Section 6.1.3), the whole object's state vector is

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_c \\ \mathbf{x}_e \end{bmatrix} = \begin{bmatrix} N_c & E_c & v_{N_c} & v_{E_c} & r(\theta_{t,1}) & r(\theta_{t,2}) & \cdots & r(\theta_{t,N_t}) \end{bmatrix}^T . \tag{6.23}$$

The angles in $\boldsymbol{\theta}_t = [\theta_{t,n}]_{n=1}^{N_t}$ correspond to a discretization of the unknown radius function r. Therefore, motivated by the vocabulary introduced in Section 5.2 about GP regression, these angles are called test angles.

Since the extent state $\mathbf{x}_e$ only consists of the radius function values at the test angles $\boldsymbol{\theta}_t$, and since these angle values are fixed, the actual state values in $\mathbf{x}_e$ should be constant for each object. These considerations would give the straightforward extent state-transition model

$$\mathbf{x}_{e,k} = \mathbf{x}_{e,k-1}. \tag{6.24}$$

However, this model does not take into account the uncertainty in the GP model as

$$\mathbf{x}_e = [r(\theta_{t,1}), \ldots, r(\theta_{t,N_t})]^T \sim \mathcal{N}(\mathbf{0}, \mathbf{K}(\boldsymbol{\theta}_t, \boldsymbol{\theta}_t)). \tag{6.25}$$

In addition, it does not have process noise.

Therefore, the model (6.24) is modified by introducing a forgetting factor as done in [37] and [20]. This yields

$$\mathbf{x}_{e,k} = \mathbf{F}_e \mathbf{x}_{e,k-1} + \mathbf{w}_{e,k} , \qquad \mathbf{w}_{e,k} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_e) , \tag{6.26}$$

where

$$\mathbf{F}_e = e^{-\tau T} \mathbf{I}_{N_t} , \tag{6.27a}$$

$$\mathbf{Q}_e = (1 - e^{-2\tau T}) \mathbf{K}(\boldsymbol{\theta}_t, \boldsymbol{\theta}_t) , \tag{6.27b}$$

$\tau > 0$ is the forgetting factor and T is the constant time step.

Therefore, the transition model for the whole state vector $\mathbf{x}$ is the combination of the extend GP model with forgetting factor (6.26) and the CV2 model with no sideslip(6.18), i.e.,

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{F}_c & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_e \end{bmatrix} \mathbf{x}_{k-1} + \mathbf{w}_k , \qquad \mathbf{w}_k \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{Q}_c & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_e \end{bmatrix} \right) , \tag{6.28}$$

where the matrices $\mathbf{F}_c$ and $\mathbf{Q}_c$ are defined in (6.19). In particular, we observe that this state-transition model is linear, and it can be directly used in the prediction step formulas of Theorem 4.4.3.

6.2.2 Lidar measurement model

Lidar sensor model

A lidar ("light detection and ranging") sensor is a high-resolution sensor that measures the distance to the nearest objects in its surroundings as a function of direction. This is achieved by sending pulsed laser light in several directions covering a region of space. These laser beams propagate through the atmosphere until they hit an object within the sensor's range. The reflected beams are received back at the sensor, and are used to estimate the distance between the sensor and the hit object, which is known as range.

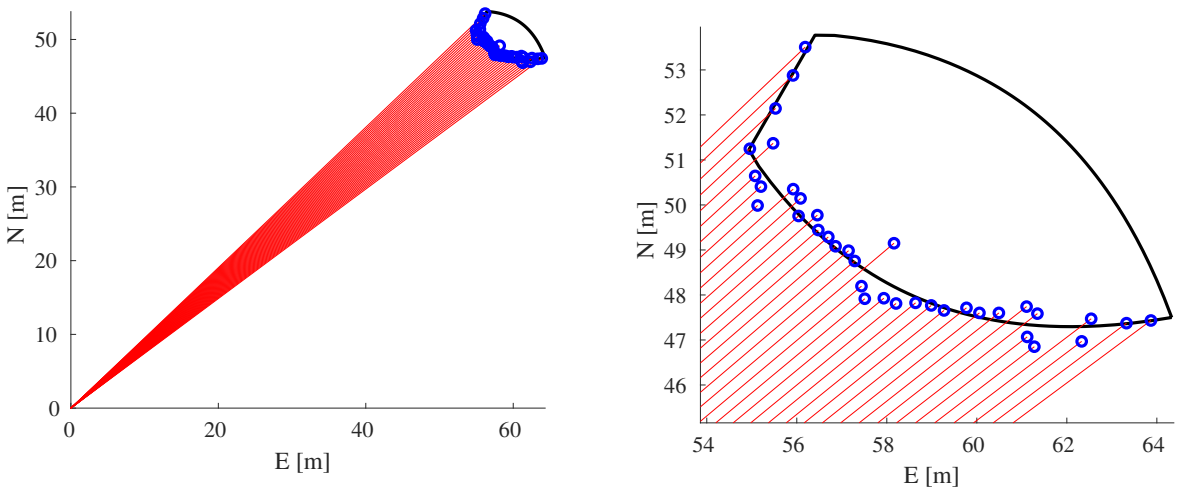
In general, the lidar range measurement are given as a function of direction, which is parameterized using azimuth and elevation angles. This results in a three dimensional point cloud of measurements. Therefore, the elevation angle is irrelevant, and the range measurements are only a function of the azimuth angle, which is the angle between the laser light beam and the North direction.

Let r_i denote the range measurement for a azimuth angle φ_i. It is assumed that the azimuth angles are known exactly, and that the range measurements can be modelled as

$$r_i = r_{i,\text{true}} + w_r \tag{6.29a}$$

$$w_r \sim \mathcal{N}(0, \sigma_r^2)\,, \tag{6.29b}$$

where $r_{m,\text{true}}$ is the true range value, w_r is the range measurement noise and $\sigma_r > 0$ is the range measurement noise strength. In other words, the range measurements are affected by unbiased normally distributed noise, whose strength is the same for all azimuth angles.



(a) Overview.

(b) Close-up.

Figure 6.3: Example of lidar measurements: laser beams (red), object boundary (black) and points that correspond to lidar range measurements (blue).

Furthermore, we assume that the lidar sensor is placed at the origin of the world reference. Figure 6.3 shows an example of simulated lidar measurements using $\sigma_r = 0.1\,\text{m}$.

Lidar measurement model using Gaussian processes

Let $\mathbf{z}$ be the coordinates in the world frame of a lidar measurement. Under the assumption that there is no measurement noise, these coordinates correspond to a radius function value $r_i = r(\theta_i)$ for an input angle θ_i as given by

$$\mathbf{z} = \mathbf{r}_c + r_i \begin{bmatrix} \cos(\theta_i + \psi_c) \\ \sin(\theta_i + \psi_c) \end{bmatrix} , \tag{6.30}$$

where $\mathbf{r}_c = [N_c, E_c]^T$ are the coordinates of the center of movement in the world frame and ψ_c is the heading angle (see section 6.1.1). This relation is illustrated in Figure 6.4. It follows from Equation (6.30) that

$$|\mathbf{z} - \mathbf{r}_c| = r_i = r(\theta_i) \tag{6.31a}$$
$$\angle(\mathbf{z} - \mathbf{r}_c) = \theta_i + \psi_c \,, \tag{6.31b}$$

which implies that

$$\mathbf{z} - \mathbf{r}_c = \mathbf{e}(\theta_i + \psi_c) r(\theta_i) \,, \tag{6.32a}$$
$$\theta_i = \angle(\mathbf{z} - \mathbf{r}_c) - \psi_c \,, \tag{6.32b}$$

where $\mathbf{e}(\varphi) = [\cos(\varphi), \sin(\varphi)]^T$, $\varphi \in \mathbb{R}$, is the unit vector function.

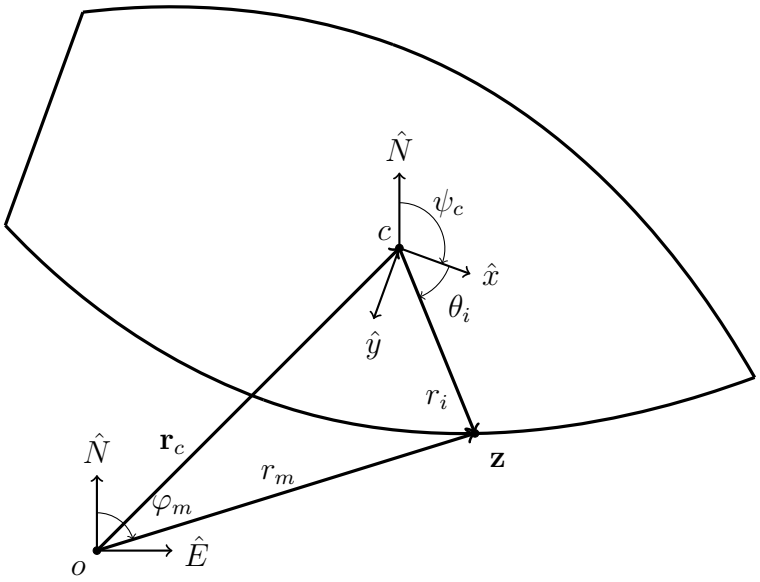


Figure 6.4: Relation between lidar measurements (range r_m for azimuth angle φ_m) and radius function measurements ($r_i = r(\theta_i)$).

Since the unknown radius function is modeled as GP of the form $r \sim \mathcal{GP}(0, k)$ with $k = k_{2\pi,a}$ (see Section 5.1.3), and since $\mathbf{x}_e = [r(\theta_{t,1}, \ldots, \theta_{t,N_t})]^T$, it follows from the GP definition (Definition 5.1.1) that

$$\begin{bmatrix} r(\theta_i) \\ \mathbf{x}_e \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} k(\theta_i, \theta_i) & \mathbf{K}(\theta_i, \boldsymbol{\theta}_t) \\ \mathbf{K}(\boldsymbol{\theta}_t, \theta_i) & \mathbf{K}(\boldsymbol{\theta}_t, \boldsymbol{\theta}_t) \end{bmatrix} \right) . \tag{6.33a}$$

Hence, by using Theorem A.1 from Appendix A, it follows that

$$r(\theta_i)|\mathbf{x}_e \sim \mathcal{N}\left(\mathbf{H}(\theta_i, \boldsymbol{\theta}_t)\mathbf{x}_e, \mathbf{R}(\theta_i, \boldsymbol{\theta}_t)\right) , \tag{6.34}$$

where

$$\mathbf{H}(\theta_i, \boldsymbol{\theta}_t) = \mathbf{K}(\theta_i, \boldsymbol{\theta}_t)[\mathbf{K}(\boldsymbol{\theta}_t, \boldsymbol{\theta}_t)]^{-1} \tag{6.35a}$$
$$\mathbf{R}(\theta_i, \boldsymbol{\theta}_t) = k(\theta_i, \theta_i) - \mathbf{K}(\theta_i, \boldsymbol{\theta}_t)[\mathbf{K}(\boldsymbol{\theta}_t, \boldsymbol{\theta}_t)]^{-1}\mathbf{K}(\boldsymbol{\theta}_t, \theta_i) , \tag{6.35b}$$

Equivalently,

$$r(\theta_i) = \mathbf{H}(\theta_i, \boldsymbol{\theta}_t)\mathbf{x}_e + \mathbf{w} , \qquad \mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}(\theta_i, \boldsymbol{\theta}_t)). \tag{6.36}$$

By inserting (6.36) into (6.32a) and adding uncorrelated measurement noise to each coordinate of the measurement $\mathbf{z}$, we obtain the measurement model equations

$$\mathbf{z} = \mathbf{r}_c + \mathbf{e}(\mathbf{z} - \mathbf{r}_c)\left[\mathbf{H}(\theta_i, \boldsymbol{\theta}_t)\mathbf{x}_e + \mathbf{w}\right] + \mathbf{w}' , \tag{6.37}$$

where

$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}(\theta_i, \boldsymbol{\theta}_t)) , \tag{6.38a}$$
$$\mathbf{w}' \sim \mathcal{N}(\mathbf{0}, \sigma_r^2\mathbf{I}_2) , \tag{6.38b}$$

σ_r is measurement noise strength and $\theta_i = \angle(\mathbf{z} - \mathbf{r}_c) - \psi_c$ according to (6.32b).

The measurement model (6.37) is highly non-linear. Therefore, it has to be linearized around the predicted state estimates in order to be used in the update step formulas of Theorem 4.4.4. The linearization formulas of (6.37) can be found in [37], and they are used the ones used in this thesis.

However, it is worth noticing that the linearization formulas in [37] assume that the input angle $\theta_i = \angle(\mathbf{z} - \mathbf{r}_c) - \psi_c$ does not depend on the measurement $\mathbf{z}$. In other words, these formulas are derived by ignoring the fact that (6.37) is actually an implicit equation for the measurement $\mathbf{z}$. Nevertheless, this approximation is reasonable if the position $\bar{\mathbf{r}}_c$ and heading $\bar{\psi}_c$ predicted estimates are good.

6.3 Gamma Gaussian inverse Wishart

The gamma Gaussian inverse Wishart (GGIW) is a state-space model that uses a random matrix model for the object's extent, and a linear measurement model that simulates radar measurements.

In the simulations performed in Chapter 7, the GGIW model is combined with a CV2 model with no sideslip for the kinematic states.

6.3.1 Random Matrix extend model

The Random matrix approach in two spatial dimensions models the object's extent as an ellipsis with center $\mathbf{r}_c = [N_c,\, E_c]^T$. Therefore, the extent can be characterized by the 2-by-2 positive definite matrix $\mathbf{X}$ that gives the implicit equations for the ellipsis in the world frame, i.e.,

$$\begin{bmatrix} N - N_c \\ E - E_c \end{bmatrix}^T \mathbf{X}^{-1} \begin{bmatrix} N - N_c \\ E - E_c \end{bmatrix} = 1. \tag{6.39}$$

The matrix $\mathbf{X}$ is called the random matrix, and the concatenation of its columns gives the extend state $\mathbf{x}_e$.

The random matrix $\mathbf{X}$ can also be represented as

$$\mathbf{X} = \begin{bmatrix} \cos(\psi_c) & -\sin(\psi_c) \\ \sin(\psi_c) & \cos(\psi_c) \end{bmatrix} \begin{bmatrix} a^2 & 0 \\ 0 & b^2 \end{bmatrix} \begin{bmatrix} \cos(\psi_c) & -\sin(\psi_c) \\ \sin(\psi_c) & \cos(\psi_c) \end{bmatrix}^T , \tag{6.40}$$

where a and b are the semi-axes of the ellipsis, which lie on the $\hat{x}$ and $\hat{y}$ axes of the body frame (see Section 6.1.1), respectively.

There are several state-transition models for the random matrix $\mathbf{X}$ in the literature, each with its own characteristics and applications (see for example [21], [12], [24] or [16]). Here, Granström's model introduced in [16] and used in [14] is chosen. This model is

$$\mathbf{X}_k = \mathbf{R}(\Delta\psi_{c,k})\mathbf{X}_{k-1}\mathbf{R}(\Delta\psi_{c,k})^T + \mathbf{W}_k \,, \tag{6.41}$$

where

$$\mathbf{R}(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix} , \tag{6.42}$$

$\Delta\psi_{c,k}$ is an approximation of the heading angle variation between the time points T_{k-1} and T_k based on the kinematic state vector, and $\mathbf{W}_t$ is the process noise, which is distributed as a Wishart distribution.

The matrix $\mathbf{R}(\Delta\psi_{c,k})\mathbf{X}_k\mathbf{R}(\Delta\psi_{c,k})^T$ in (6.41) represents the ellipsis given by $\mathbf{X}_k$ rotated by an angle $\Delta\psi_{c,k}$, while the term $\mathbf{W}_t$ accounts for approximation errors and uncertainties in the state-transition.

See [16] for the details about this extend state-transition model. In any case, it is important to note that the model (6.41) cannot be used in conjunction with Theorem 4.4.3 since the Wishart distribution is not a multivariate normal distribution. Nevertheless, PMBM prediction and update laws for the GGIW model can be found in [14], and they have been implemented in the code by Yuxuan Xia in [41].

6.3.2 Radar measurement model

The world frame coordinates of a radar measurement $\mathbf{z}$ are distributed as

$$l(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\begin{bmatrix} N_c \\ E_c \end{bmatrix}, \mathbf{X} + \sigma_r \mathbf{I}_2\right), \tag{6.43}$$

where σ_r is the measurement noise strength.

Equivalently,

$$\mathbf{z} = \begin{bmatrix} N_c \\ E_c \end{bmatrix} + \mathbf{v}_t\,, \qquad \mathbf{v}_t \sim \mathcal{N}\left(\mathbf{0},\, \mathbf{X} + \sigma_r \mathbf{I}_2\right) \tag{6.44}$$

In other words, the single object-generated measurement $\mathbf{z}$ is normally distributed with the center of the elliptical extent $[N_c,\, E_c]^T$ as its mean value and with $\mathbf{X} + \sigma_r \mathbf{I}_2$ as its covariance matrix.

Chapter 7

Simulations and results

The PMBM filter for EOT derived in Chapter 4 is integrated with the state-space models presented in Section 6.2 and Section 6.3, and the resulting methods are tested under different simulations.

The implementation of the filter was done in Matlab, and it is based on the code by Yuxuan Xia, which can be found on Github (see [41]). This PMBM code implements the theory developed in [42] and [18]. In particular, the GGIW model (Section 6.3) is integrated with the PMBM filter by default. Therefore, this code was modified to integrate the GP and lidar model presented in Section 6.2, among other minor modifications.

The PMBM-filter parameters that are used for all simulations are a constant survival probability of $p^s = 0.99$ and a constant detection probability of $p^d = 1$. Furthermore, in order to make the filter computationally tracktable, the gating threshold is set to 0.99, the thresholds for the existence probability of the Bernoulli components and for the weights in both the PPP and the MBM is set to 10^{-2}, while the maximum number of MBM components is set to 100 (see Section 4.5).

7.1 Random walk simulation

In the random walk simulations, the kinematic states of the objects are simulated according to the CV2 model with no sideslip (see Section 6.1.3) with a process noise strength of $\sigma_w = 0.05\,\mathrm{m\,s^{-1}}$. According to [40], this level of process noise corresponds to a slowly varying maritime vessel movement.

The time step for all random walk simulations is constant and equal to $1\,\mathrm{s}$, and the simulations last for one minute. Moreover, these simulations take place in a region whose coordinates extend from 0 to $200\,\mathrm{m}$ in the North direction and from -150 to $150\,\mathrm{m}$ in the East direction. In this region, objects are generated at two specific birth points located at $[100, -125]\mathrm{m}$ and $[100, 125]\mathrm{m}$. At the initial time point there are two objects, each located at either birth point, and from the next time point onwards, new objects are generated according to a Poisson distribution with rate $\lambda^b = 0.01\,\mathrm{s^{-1}}$.

The objects generated at the west birth point represent small boats (e.g. kayaks) that move to the East with a velocity of approximately $1\,\mathrm{m\,s^{-1}}$, while the object generated at the east birth point represent medium sized boats (e.g. skiffs) that move to the West with a velocity of approximately $3\,\mathrm{m\,s^{-1}}$, which is almost 6 knots.

More precisely, the initial kinematic state vector is given according to (6.16). Hence,

$$\mathbf{x}_{c,0} = \bar{\mathbf{x}}_{c,0} + \mathbf{w}_1 \tag{7.1a}$$

$$\mathbf{w}_1 \sim \mathcal{N}\left(\mathbf{0}, \sigma_c^2 \begin{bmatrix} 1 & \frac{1}{T} \\ \frac{1}{T} & \frac{2}{T^2} \end{bmatrix}\right), \tag{7.1b}$$

where $\bar{\mathbf{x}}_{c,0}$ is the corresponding initial predicted state vector, which is $\bar{\mathbf{x}}_{c,0} = [100,\ -125,\ 0,\ 1]^T$ if the object originated at the west birth point or $\bar{\mathbf{x}}_{\mathbf{c,0}} = [100,\ 125,\ 0,\ -3]^T$ if it originated at the east birth point. Moreover, σ_c is the strength of the initial prediction error, which is set to $0.5\,\mathrm{m}$.

The extend characteristics of the objects vary depending on the used state-transition and measurement models.

7.1.1 Results using the GGIW model and discussion

As presented in Section 6.3.1, the random matrix approach assumes the extents to be ellipses. In the simulations, all objects originated at the west birth point have semiaxes of $3\,\mathrm{m}$ and $0.5\,\mathrm{m}$, while all objects originated at the west birth point have semiaxes of $6\,\mathrm{m}$ and $2.5\,\mathrm{m}$. However, in any case, the initial predicted extend is taken as a circle of radius $5\,\mathrm{m}$.

Regarding the simulated radar measurements (see Section 6.3.2), the strength of the additional measurement noise is set to $\sigma_r = 0.1\,\mathrm{m}$, and the number of object-generated measurements and clutter are sampled according to Poisson distributions with rates $\lambda^m = 15$ and $\lambda^c = 30$, respectively. Moreover, the spatial coordinates of the clutter measurements are sampled according to an uniform distribution on the simulation region.

Figure 7.1 illustrates the scenario of a random walk simulation. Figure 7.1a shows the trajectories of all the spawned objects. Note that in this particular simulation two extra objects were generated in addition to the two initial ones, and that the trajectories resemble vessels moving mostly forward. Moreover, the simulated object-generated measurements and clutter seem realistic as shown in Figure 7.1b.

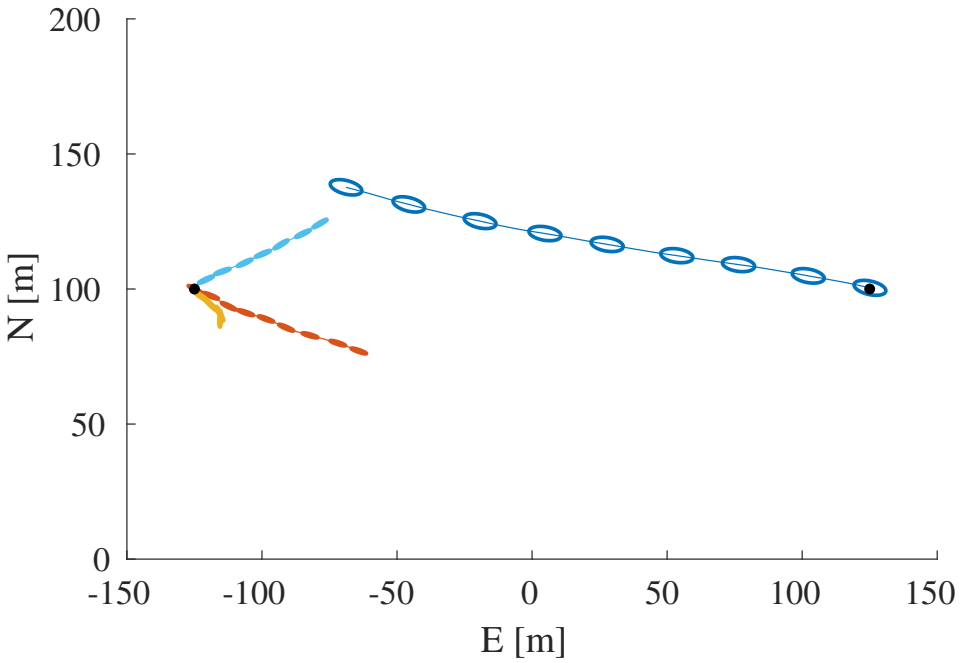
The same parameter values used to simulate the state trajectories and the measurements are employed by the PMBM filter to track the objects. In other words, no additional parameter tuning was performed.

Several simulations were undertaken, and the results are quite similar. In general, objects are detected at the same time point they are generated, and the obtained state estimates are accurate.

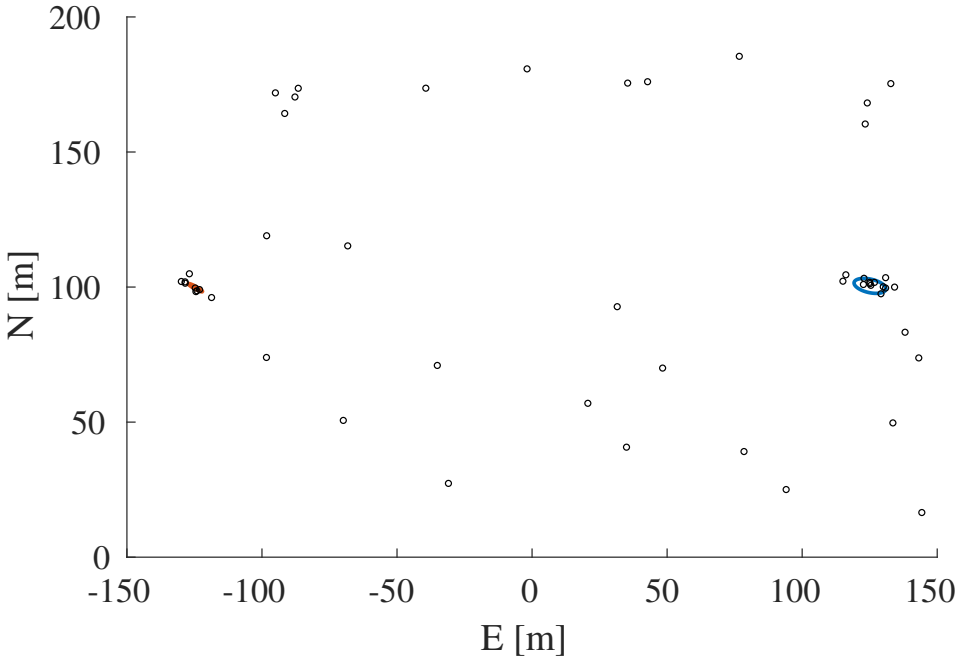
As a representative example, the results for the two first generated objects in the simulation illustrated in Figure 7.1 are shown in Figures 7.2 to 7.5. Figure 7.2 and Figure 7.4 show the true

and estimated kinematic state values for the objects generated at the west and east birth point, respectively, while Figure 7.3 and Figure 7.5 show the corresponding time-lapses for their extend trajectories and estimates.

As one can observe in Figure 7.2 and Figure 7.4, the position estimates are very accurate, while the velocity and heading estimates are not that good. Nevertheless, a good estimate of the extend trajectories is achieved as shown in Figure 7.3 and Figure 7.5.

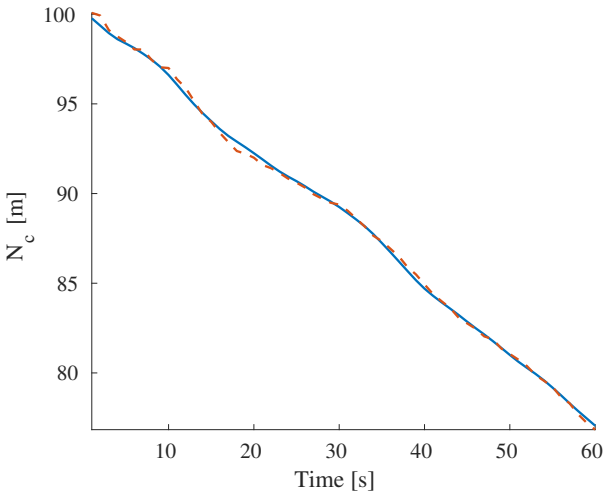


(a) Simulated trajectories for the different objects with a time-lapse of their extent positions. The black dots mark the points where objects are generated.

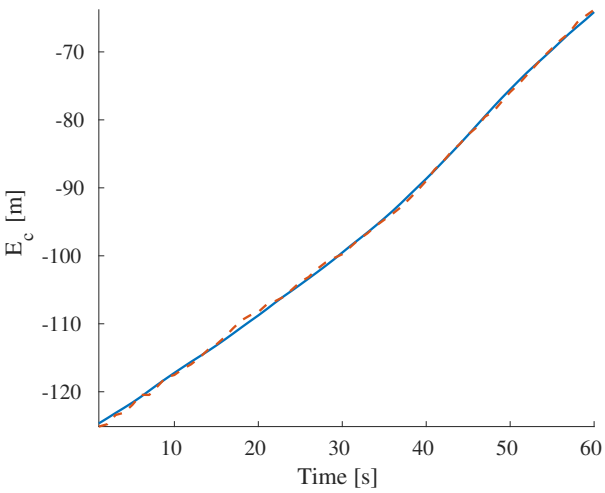


(b) Simulation at time $t = 0\,\mathrm{s}$: The two initial objects (blue and red) and measurements.

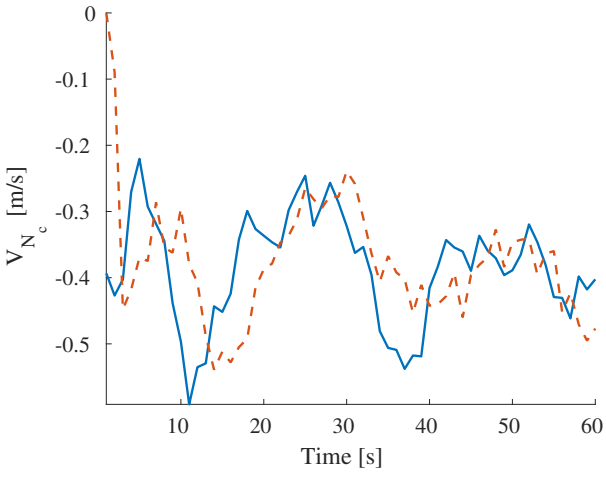
Figure 7.1: Random walk simulation and simulated radar measurements.



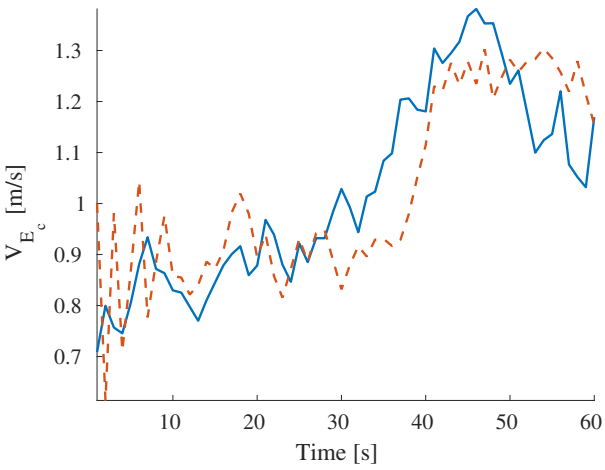
(a) North coordinate N_c.



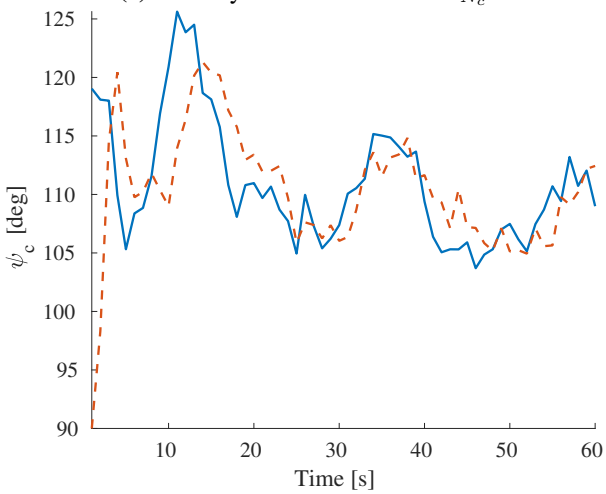
(b) East coordinate E_c.



(c) Velocity in North direction v_{N_c}.

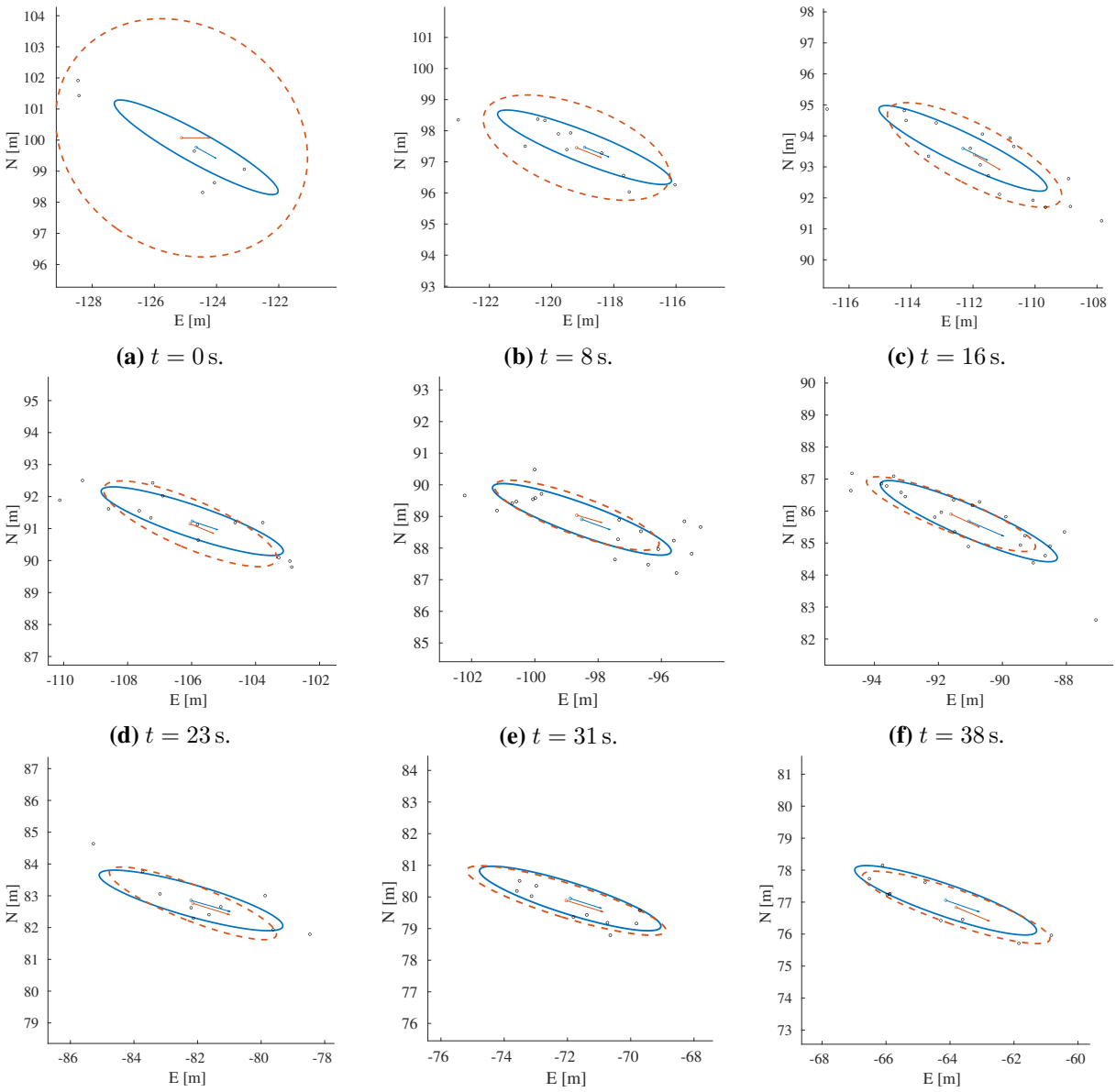


(d) Velocity in East direction v_{E_c}.



(e) Heading ψ_c.

Figure 7.2: Results for the kinematic states using the PMBM filter for MEOT and the GGIW model: Actual state values (blue) and corresponding state estimates (red dashed) of the first object generated at the West.

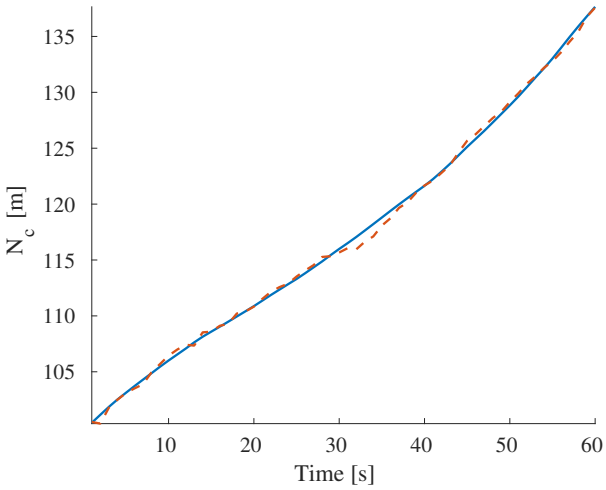


(a) $t = 0\,\mathrm{s}$. **(b)** $t = 8\,\mathrm{s}$. **(c)** $t = 16\,\mathrm{s}$.

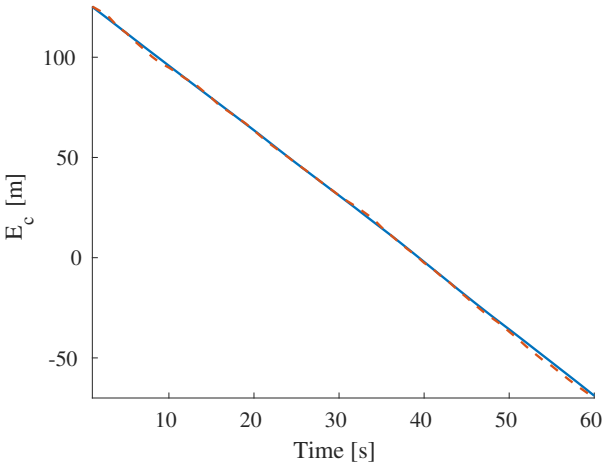
(d) $t = 23\,\mathrm{s}$. **(e)** $t = 31\,\mathrm{s}$. **(f)** $t = 38\,\mathrm{s}$.

(g) $t = 45\,\mathrm{s}$. **(h)** $t = 53\,\mathrm{s}$. **(i)** $t = 60\,\mathrm{s}$.

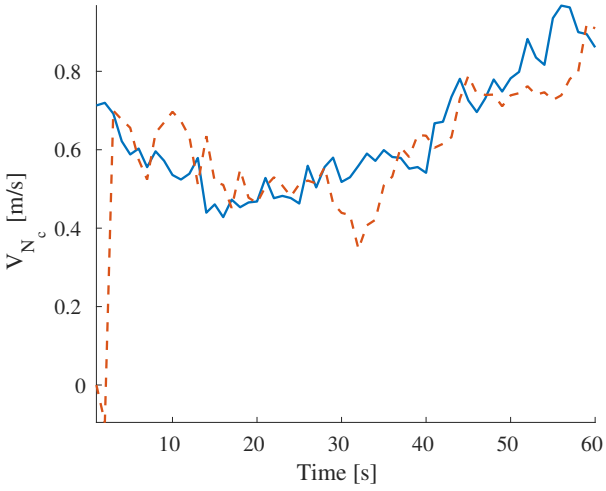
Figure 7.3: Time-lapse for the first object generated at the West using the PMBM filter for MEOT and the GGIW model: Actual extend (blue ellipsis), estimated extend (red dashed ellipsis), actual velocity vector (blue arrow), estimated velocity vector (red arrow) and radar measurements (black dots).



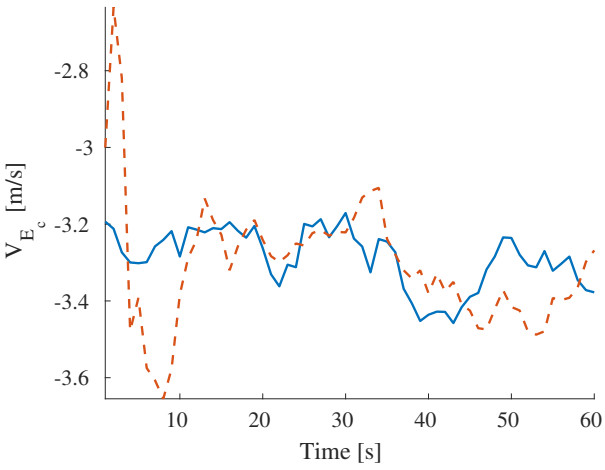
(a) North coordinate N_c.



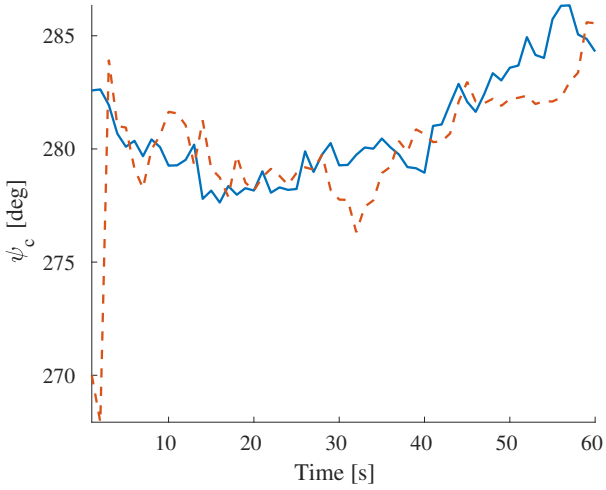
(b) East coordinate E_c.



(c) Velocity in North direction v_{N_c}.

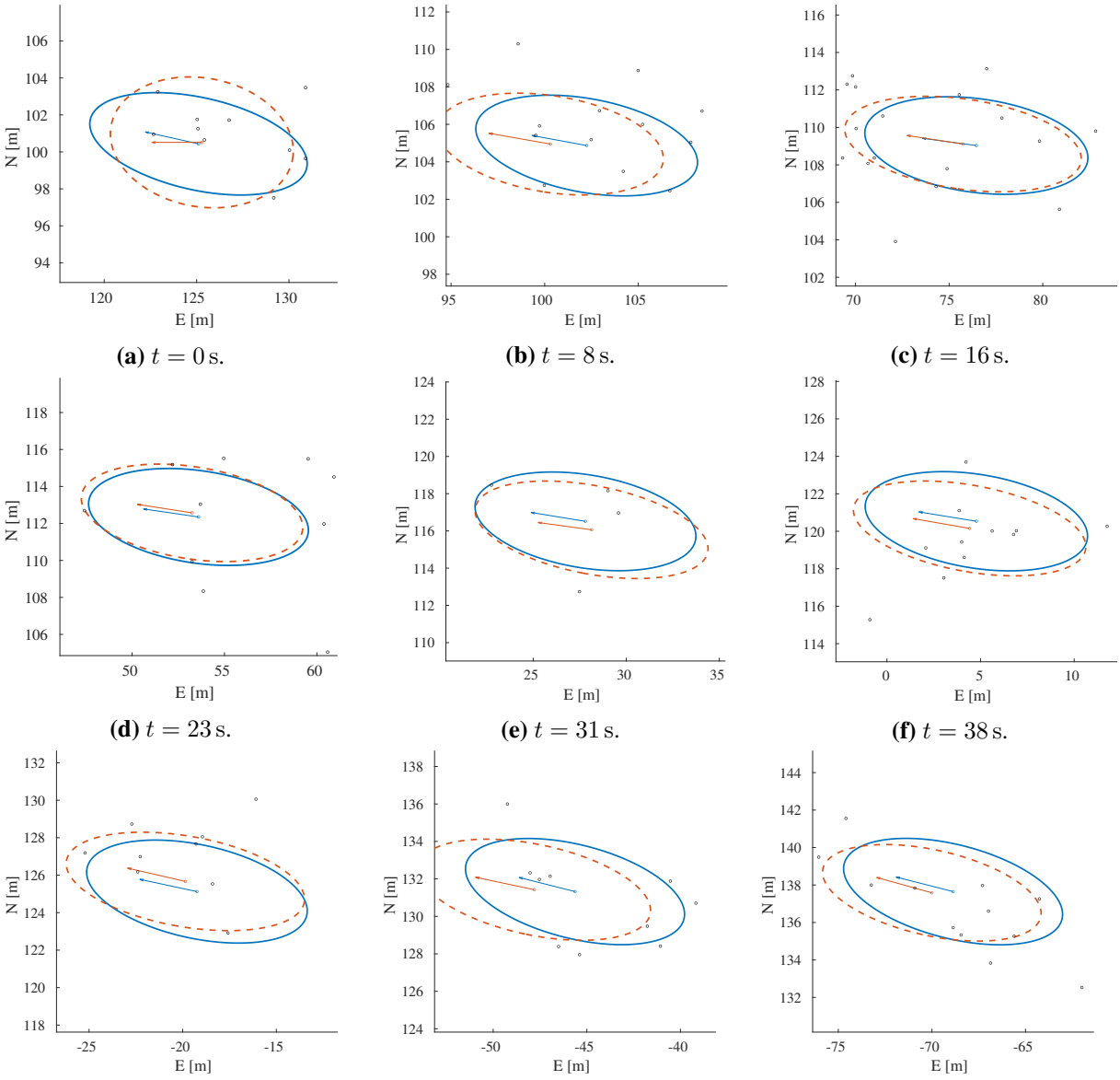


(d) Velocity in East direction v_{E_c}.



(e) Heading ψ_c.

Figure 7.4: Results for the kinematic states using the PMBM filter for MEOT and the GGIW model: Actual state values (blue) and corresponding state estimates (red dashed) of the first object generated at the East.



(a) $t = 0\,\mathrm{s}$. **(b)** $t = 8\,\mathrm{s}$. **(c)** $t = 16\,\mathrm{s}$.

(d) $t = 23\,\mathrm{s}$. **(e)** $t = 31\,\mathrm{s}$. **(f)** $t = 38\,\mathrm{s}$.

(g) $t = 45\,\mathrm{s}$. **(h)** $t = 53\,\mathrm{s}$. **(i)** $t = 60\,\mathrm{s}$.

Figure 7.5: Time-lapse for the first object generated at the East using the PMBM filter for MEOT and the GGIW model: Actual extend (blue ellipsis), estimated extend (red dashed ellipsis), actual velocity vector (blue arrow), estimated velocity vector (red arrow) and radar measurements (black dots).

7.1.2 Results using the GP and lidar model and discussion

As GPes allow to model a wide variety of extends (see Section 5.1), the extends of the objects generated at the east birth point are modeled to resemble the hull of a medium sized boat as illustrated in Figure 7.6. Each of these hulls is symmetric about its longitudinal axis, and it has an overall length of $L = 10\,\mathrm{m}$ and a beam of $B = 5\,\mathrm{m}$. The width of the beam is achieved at a distance $D = 6\,\mathrm{m}$ from the bow. Furthermore, the hull ends in a flat stern of width $S = 3\,\mathrm{m}$. With these parameter values, the dimensions of these extends are roughly the same as the elliptical extends generated at the east birth point in the simulations of Section 7.1.1.

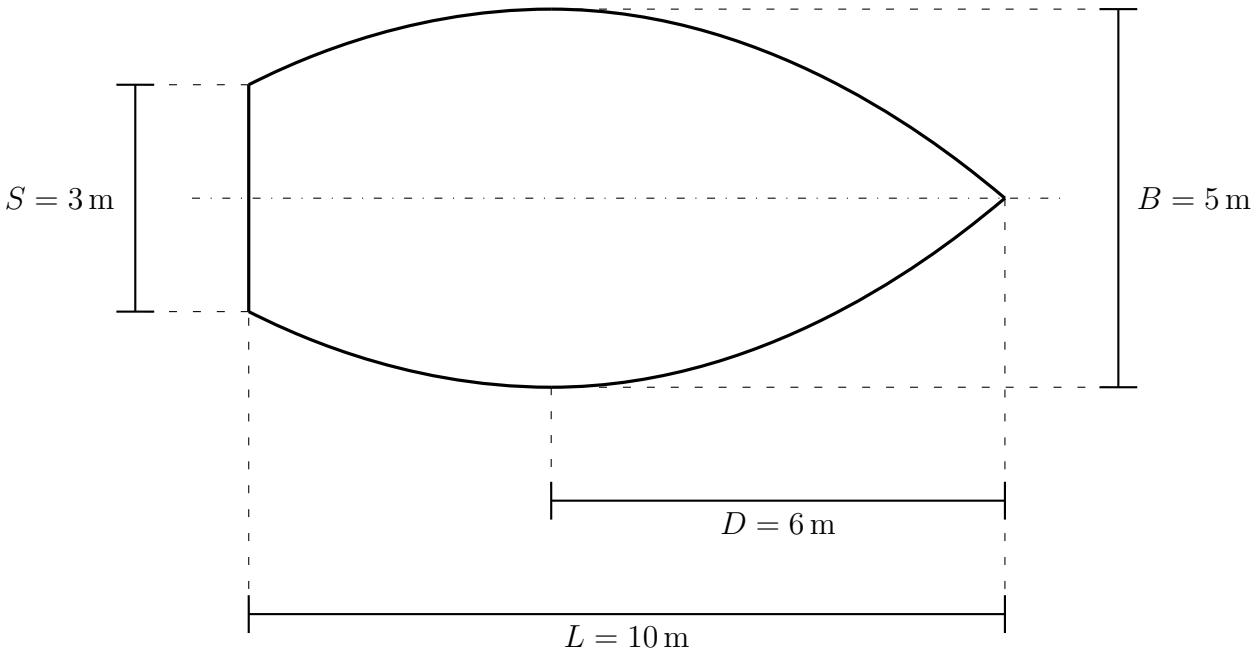


Figure 7.6: Extent of the objects that are generated at the east birth point.

On the other side, the extends generated at the west birth point are ellipses with semiaxes of $3\,\mathrm{m}$ and $0.5\,\mathrm{m}$ as in Section 7.1.1. Furthermore, the initial predicted extends are also taken as a circle of radius $5\,\mathrm{m}$.

The lidar sensor is located at the origin of the world coordinate system, and it has a bearing angle resolution of $0.2°$ and a range measurement noise strength of $\sigma_r = 0.1\,\mathrm{m}$. These parameter values are inspired by the technical specifications of the Velodyne's Puck (VLP-16) lidar sensor (see [5]). In addition to the object-generated measurements simulated using these lidar parameters, clutter is simulated according to a Poisson distribution with rate $\lambda^c = 30$ for the number of clutter measurements, and a uniform distribution on the simulation region for their spatial coordinates, as done in Section 7.1.1.

In the simulations, the same parameter values used to simulate the kinematic state trajectories and the measurements are employed by the PMBM filter to track the objects. In other words, no additional tuning of the parameters of the CV2 model and the lidar measurement model was performed. However, the GP hyperparameters σ_f, σ_b, σ_n, l and the forgetting factor τ were tuned.

Model selection is the problem of choosing a GP model for a particular application, i.e. choosing a mean and covariance function, as well as determining hyperparameter values. In general, model selection is far from a trivial task. The main reason for this is that GPes are non-parametric models. Therefore it may not be obvious which GP model family or parameter values should be chosen. In addition, some covariance functions may depend on many hyperparameters [30, p.105-106]. However, depending on the particular application, the GP model family or the value of some hyperparameters may be easy to specify.

Several simulations were undertaken during the GP tuning process. Although existing objects are detected as they are generated by the filter, their extend estimates could diverge from the actual extend over time if their position or heading estimates are not accurate enough. The reason behind this is that an inaccurate position or heading estimate will associate the gated measurements to the wrong input angles in the GP model, which could lead to an incorrect extend estimate. If in addition the uncertainties in the extent estimate are thought to be much less than they actually are due to the chosen GP parameters, the extend estimate will never recover and it will probably diverge as the simulation progresses.

In order to avoid this undesirable possibility, the process noise for the GP extend model is increased by selecting relatively large values for the forgetting factor τ and the variance of the noise component of the GP covariance function σ_n. A large forgetting factor τ gives more weight to present measurements than to present estimates, which are a result of previous measurements. Moreover, by increasing the variance of the noise component σ_n, the uncertainty in the radius function estimates is increased. The combination of these two effects allow to recover from poor extent estimates given that the state estimates improve over time.

The final selected GP parameters are the round numbers:

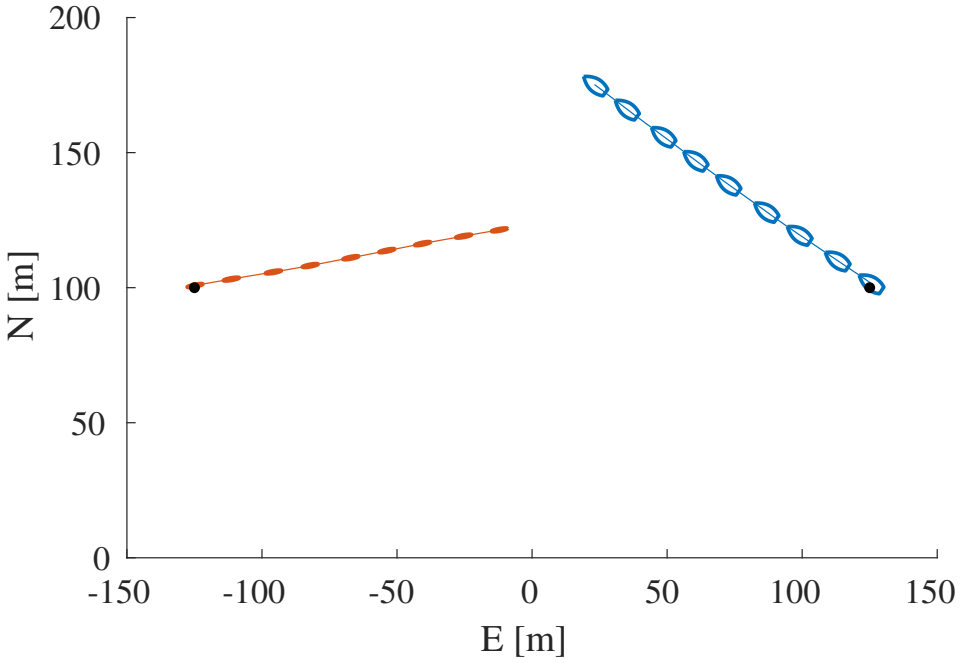
$$\sigma_f = 1\,\mathrm{m} \tag{7.2a}$$
$$\sigma_b = 5\,\mathrm{m} \tag{7.2b}$$
$$\sigma_n = 0.5\,\mathrm{m} \tag{7.2c}$$
$$l = \frac{\pi}{4} \tag{7.2d}$$
$$\tau = 10^{-2}\mathrm{s}^{-1}. \tag{7.2e}$$

The variance of the bias component σ_b is equal to the radius of the initial extend estimate, which is a rough estimate of the extent dimensions. The length-scale value states that radius function values for test angles that are more than $\frac{\pi}{4}$ apart are basically uncorrelated. The rest of parameter values in (7.2) are selected by tuning. Note that the variance of the noise component σ_n has a large value compared to the size of the extends.

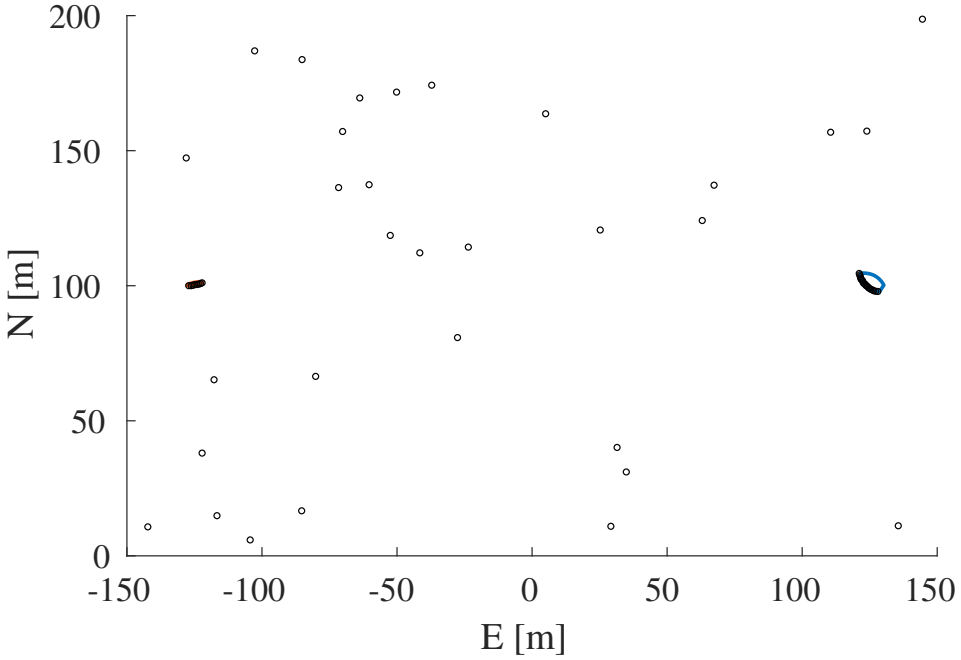
For the GP parameter values in (7.2), satisfactory simulation results are obtained. Figure 7.7 illustrates the scenario of one such simulation. Note that no extra object was generated in addition to the two initial objects. Furthermore, Figure 7.8 and Figure 7.10 show the true and estimated kinematic state values for the objects generated at the west and east birth point, respectively, while Figure 7.9 and Figure 7.11 show the corresponding time-lapses for their extend trajectories and estimates.

As one can observe in Figure 7.8 and Figure 7.10, the position estimates are very accurate, while the velocity and heading estimates are not that good, specially at the beginning of the simulation. This is reflected in the poor initial extent estimates as shown in Figure 7.9 and Figure 7.11. However, as also shown in these figures, the extend estimates become more accurate as the heading estimate improves and more measurements come in.

Note that the extend estimates are always inside their 99% confidence region, even when the estimates are poor. Although these confidence regions are relatively large due to the large value of the variance of the noise component σ_n, they are sufficient for a collision avoidance system since such a system would avoid even a large area around the tracked objects.

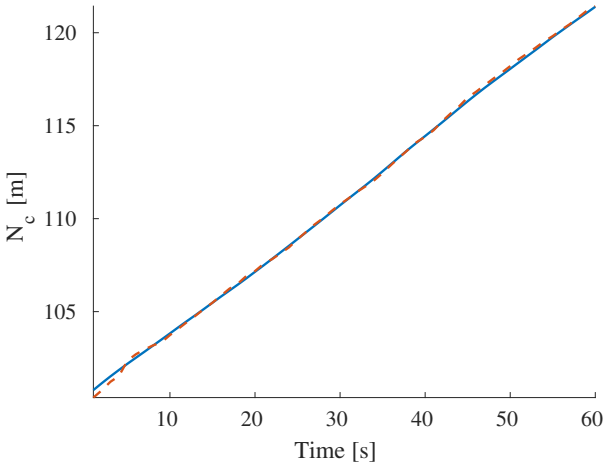


(a) Simulated trajectories for the different objects with a time-lapse of their extent positions. The black dots mark the points where objects are generated.

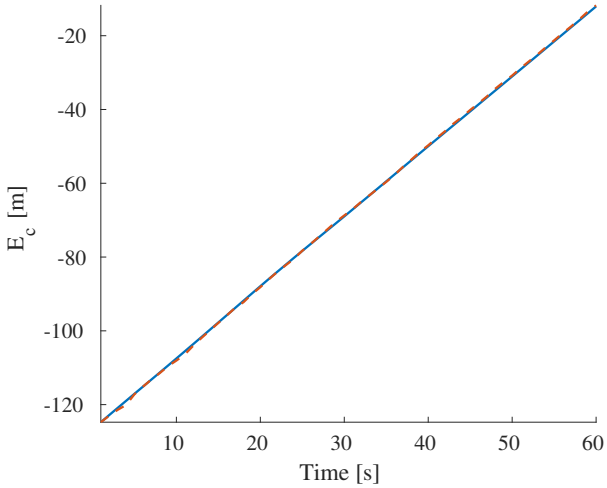


(b) Simulation at time $t = 0\,\mathrm{s}$: The two initial objects (blue and red) and measurements.

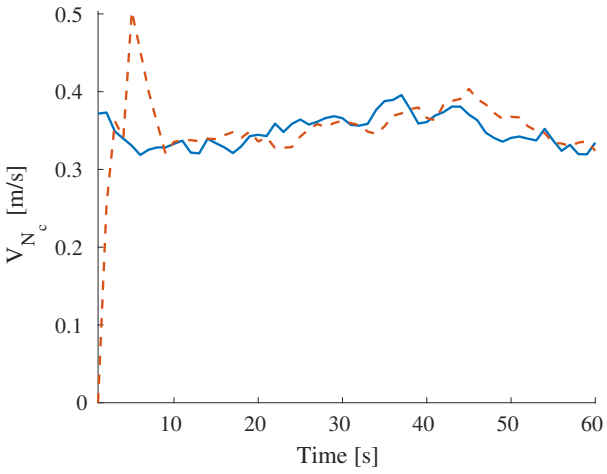
Figure 7.7: Random walk simulation and simulated lidar measurements.



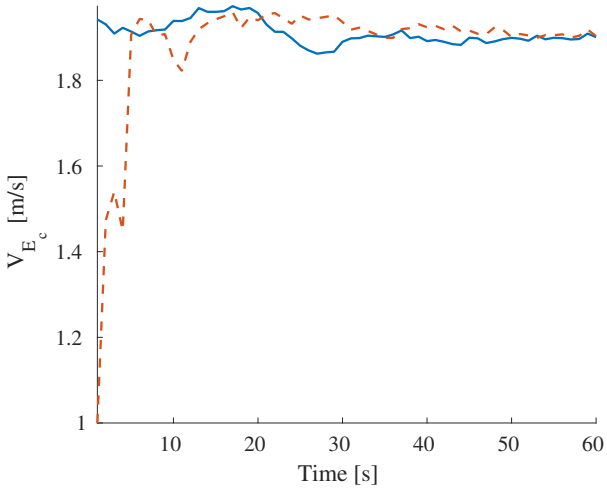
(a) North coordinate N_c.



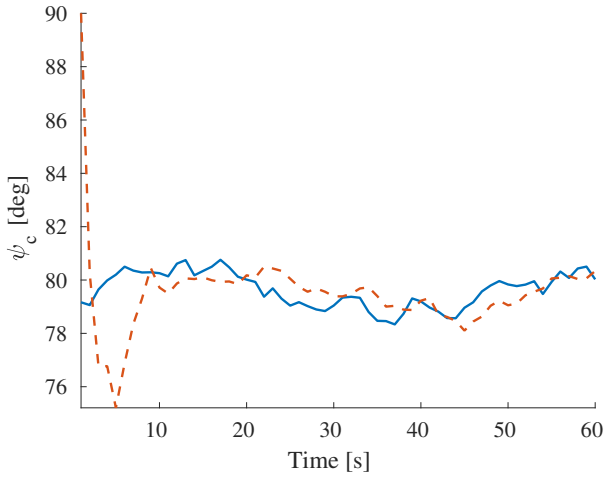
(b) East coordinate E_c.



(c) Velocity in North direction v_{N_c}.

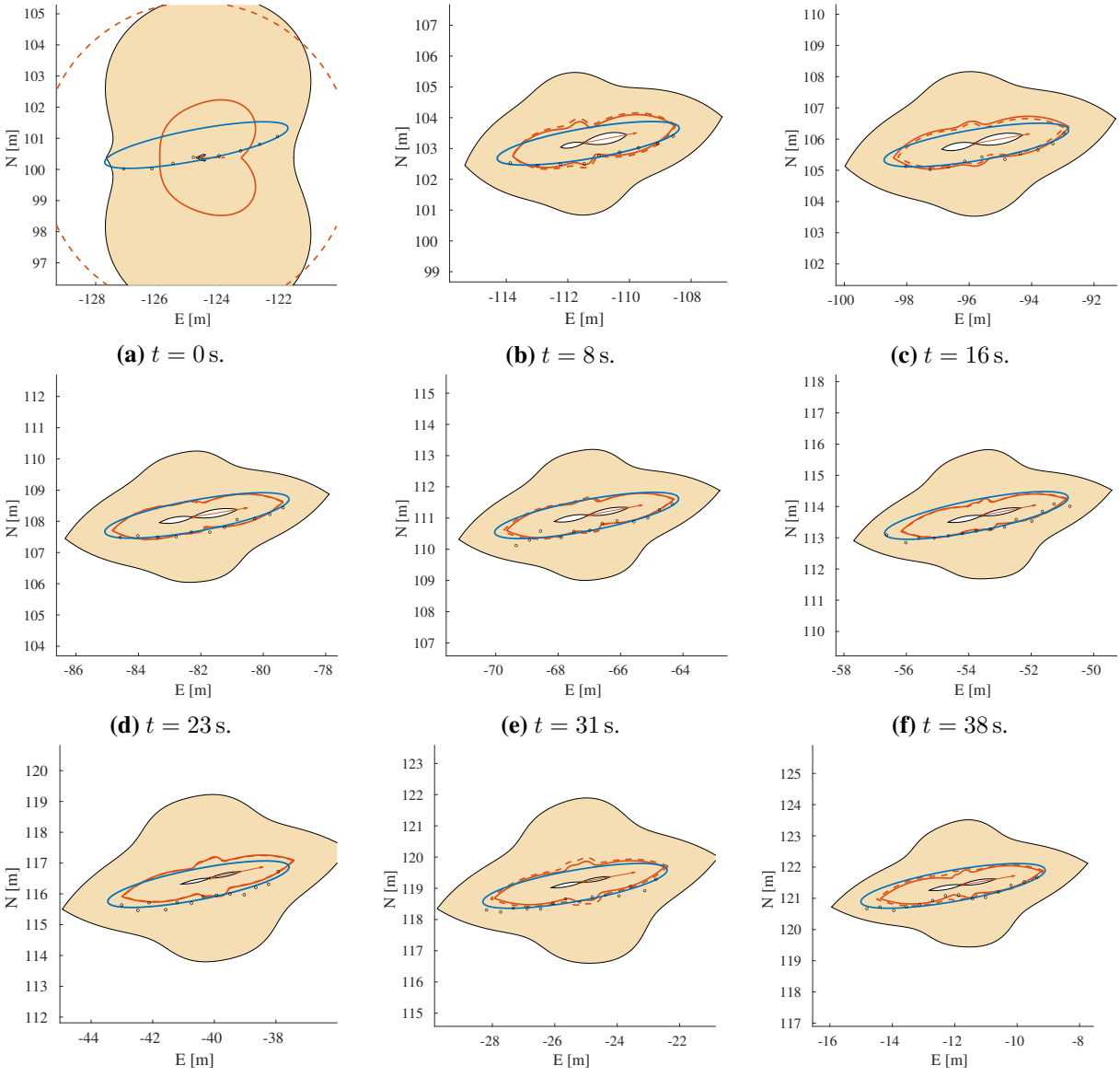


(d) Velocity in East direction v_{E_c}.



(e) Heading ψ_c.

Figure 7.8: Results for the kinematic states using the PMBM filter for MEOT and the GP and lidar model: Actual state values (blue) and corresponding state estimates (red dashed) of the first object generated at the West.

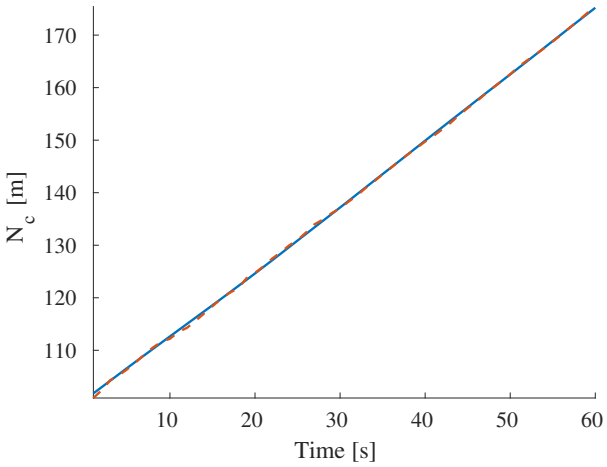


(a) $t = 0\,\mathrm{s}$. **(b)** $t = 8\,\mathrm{s}$. **(c)** $t = 16\,\mathrm{s}$.

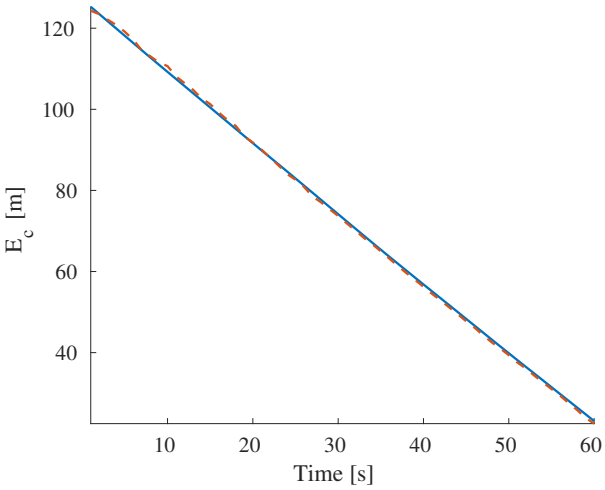
(d) $t = 23\,\mathrm{s}$. **(e)** $t = 31\,\mathrm{s}$. **(f)** $t = 38\,\mathrm{s}$.

(g) $t = 45\,\mathrm{s}$. **(h)** $t = 53\,\mathrm{s}$. **(i)** $t = 60\,\mathrm{s}$.

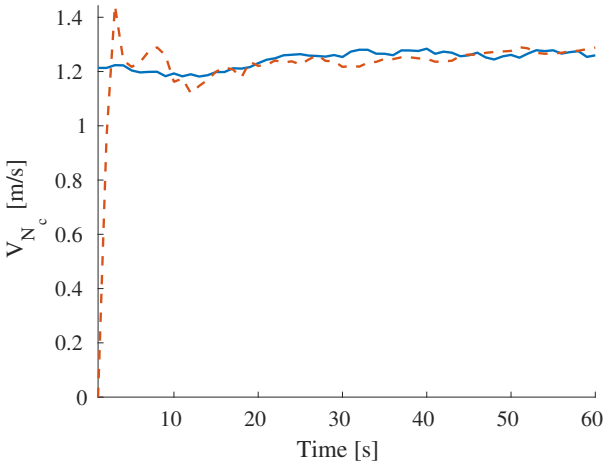
Figure 7.9: Time-lapse for the first object generated at the West using the PMBM filter for MEOT and the GP and lidar model: Actual extend (blue closed curve), predicted extend estimate (red dashed closed curve), updated extend estimate (red closed curve), 99% confidence region for the updated extend estimate (shadowed region), actual velocity vector (blue arrow), estimated velocity vector (red arrow) and lidar measurements (black dots).



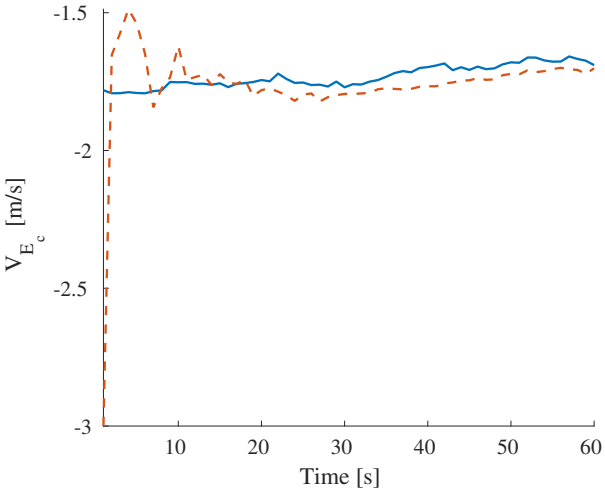
(a) North coordinate N_c.



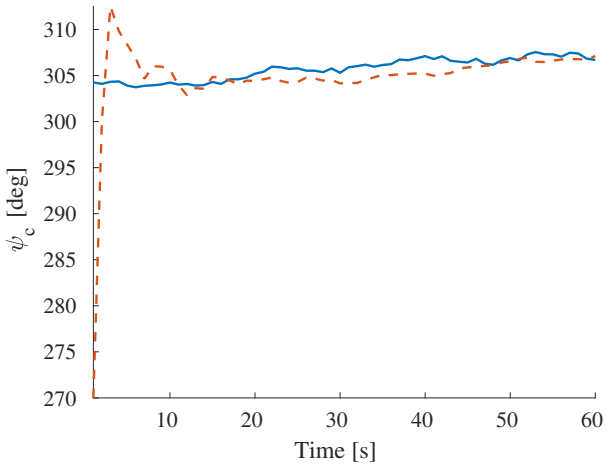
(b) East coordinate E_c.



(c) Velocity in North direction v_{N_c}.

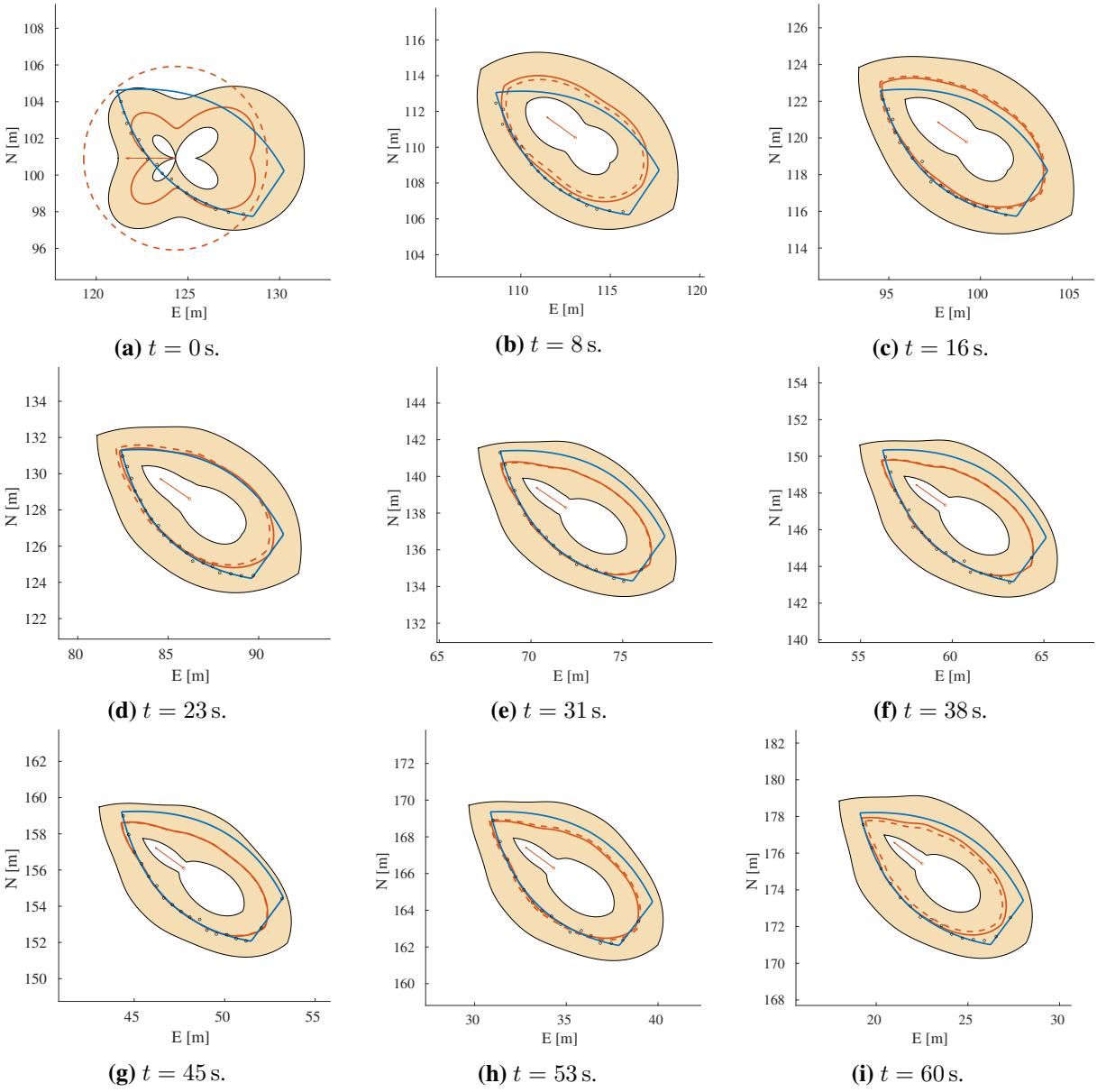


(d) Velocity in East direction v_{E_c}.



(e) Heading ψ_c.

Figure 7.10: Results for the kinematic states using the PMBM filter for MEOT and the GP and lidar model: Actual state values (blue) and corresponding state estimates (red dashed) of the first object generated at the East.



(a) $t = 0\,\mathrm{s}$. **(b)** $t = 8\,\mathrm{s}$. **(c)** $t = 16\,\mathrm{s}$.

(d) $t = 23\,\mathrm{s}$. **(e)** $t = 31\,\mathrm{s}$. **(f)** $t = 38\,\mathrm{s}$.

(g) $t = 45\,\mathrm{s}$. **(h)** $t = 53\,\mathrm{s}$. **(i)** $t = 60\,\mathrm{s}$.

Figure 7.11: Time-lapse for the first object generated at the East using the PMBM filter for MEOT and the GP and lidar model: Actual extend (blue closed curve), predicted extend estimate (red dashed closed curve), updated extend estimate (red closed curve), 99% confidence region for the updated extend estimate (shadowed region), actual velocity vector (blue arrow), estimated velocity vector (red arrow) and lidar measurements (black dots).

7.2 Coordinated turn simulation

The coordinated turn simulations are very similar to the random walk simulations (Section 7.1). The major difference is that the trajectories of the kinematic states are no longer sampled according to the CV2 model without sideslip (see Section 6.1.3). Instead, the objects follow predetermined lines and circle arcs with a constant velocity.

In addition, the simulation region is smaller than the one in (Section 7.1), extending from 0 to $150\,\mathrm{m}$ in the North direction and from -100 to $100\,\mathrm{m}$ in the East direction, and the simulations last for $90\,\mathrm{s}$.

There are also two birth points for objects, which are located at $[100, -75]\mathrm{m}$ and $[20, 75]\mathrm{m}$, respectively. The objects generated at the west birth point take a turn to starboard describing a circle of radius $200\,\mathrm{m}$ with a velocity of $1\,\mathrm{m\,s^{-1}}$. On the other hand, the objects generated at the east birth point sail first to the West for $100\,\mathrm{m}$, then turn to port and describe a semicircle of radius $50\,\mathrm{m}$, and finally continue back to the East. These trajectories are described with a velocity of $3\,\mathrm{m\,s^{-1}}$. In addition, the kinematic state trajectory of each object is perturbed with a small amount of random white noise w_k, which is distributed according to (6.13) as

$$\mathbf{w}_k \sim \mathcal{N}\left(\mathbf{0}, \sigma_w^2 \begin{bmatrix} \frac{T^2}{3} & \frac{T}{2} \\ \frac{T}{2} & 1 \end{bmatrix}\right), \tag{7.3}$$

where $\sigma_w = 0.05\,\mathrm{m\,s^{-1}}$.

With the exception of the above-mentioned differences, the rest of simulation parameters are the same as in Section 7.1.

Note that the coordinated turn trajectories do not correspond to a CV2 model without sideslip (see Section 6.1.3). However, similar trajectories can perfectly be described by maritime vessels. Therefore, the objective of these simulations is to test the robustness of the implemented filter when using a CV models for the kinematic states.

7.2.1 Results using the GGIW model and discussion

The parameter values for the extends and the measurement model are the same as in Section 7.1.1, with the exception of the process noise strength σ_w used in the filter, whose value has to be considerably larger than the one in Section 7.1.1 due to the kinematic model discrepancies.

In order to do not lose track of the objects, the process noise strength σ_w was increased to the round number $1\,\mathrm{m\,s^{-1}}$. Such a process noise strength is relatively large in the context of maritime vessels according to [40], where a value of around $0.5\,\mathrm{m\,s^{-1}}$ is considered to represent a vessel with a highly-varying movement.

Figure 7.12 illustrates the scenario for one simulation. Figure 7.13 and Figure 7.15 show the true and estimated kinematic state values for the objects generated at the west and east birth point, respectively, while Figure 7.14 and Figure 7.16 show the corresponding time-lapses for their extend trajectories and estimates.

As one can observe in Figure 7.13 and Figure 7.15, the kinematic state estimates are poor, specially the velocity and heading estimates. The reason for this poor performance is the kinematic model used for estimation, which despite its large process noise still tries to fit a CV model. This causes the estimated kinematic state values to oscillate around their corresponding true values. Although the dimensions of the estimated extents are for the most part acceptable (see Figure 7.14 and Figure 7.16), the poor position and velocity estimates causes the extend to be placed far away from its actual position. This offset may cause the filter to lose track of the object, as happens with the object generated at the west birth point towards the end of the simulation as shown in Figure 7.14.

Further tuning of the process noise strength could lead to a better performance for these particular simulations. However, considerably sharper turns that the ones simulated here are possible for vessels of this size, and just increasing the process noise strength is not a sustainable strategy since this would lead to large uncertainties in the estimates.

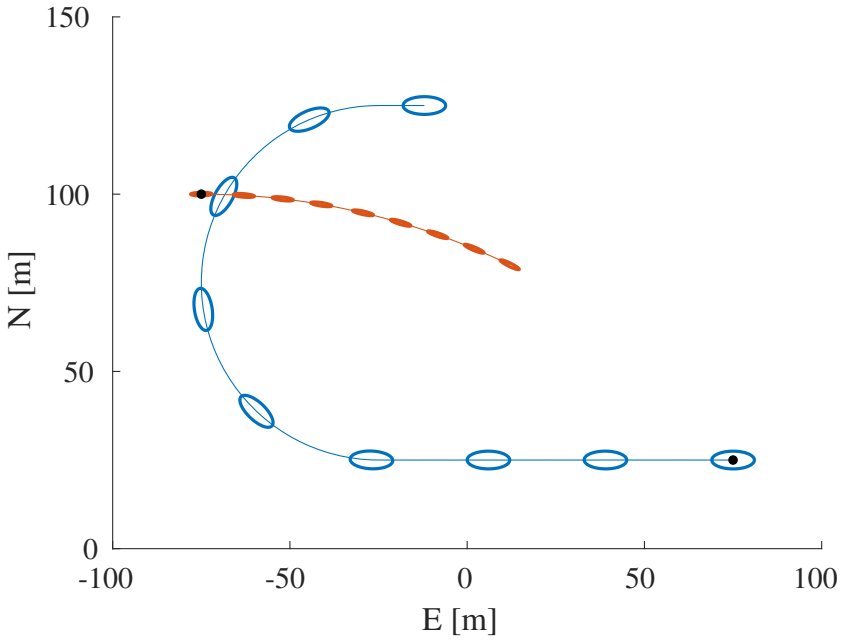
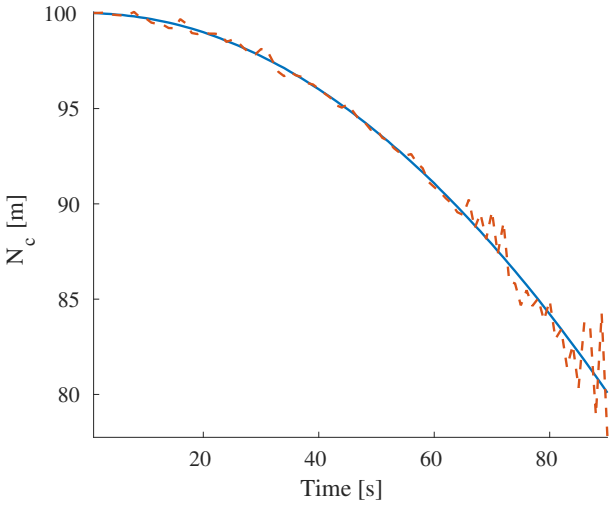
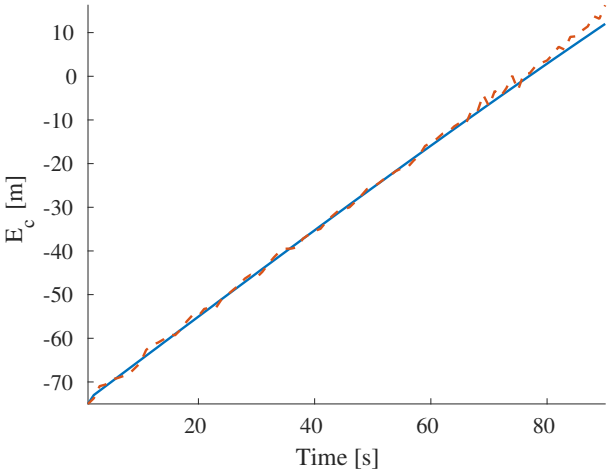


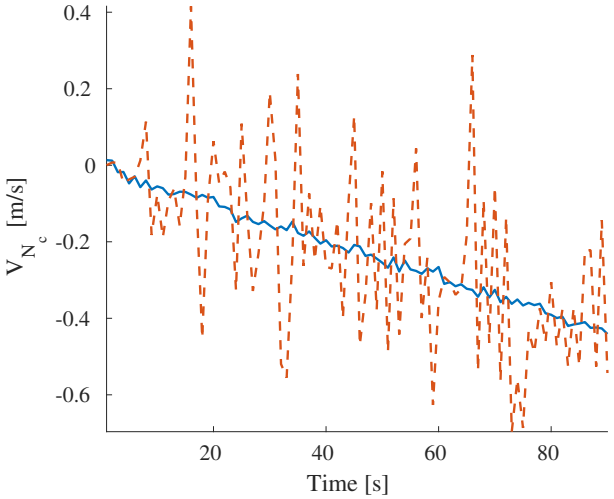
Figure 7.12: Coordinated turn simulation: Simulated trajectories for the different objects with a time-lapse of their extent positions. The black dots mark the points where objects are generated.



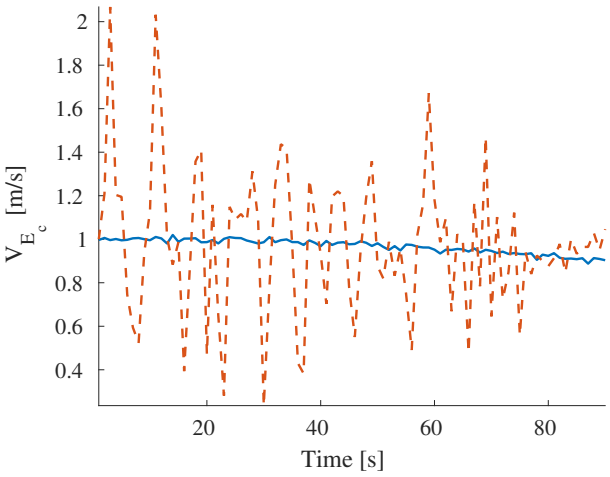
(a) North coordinate N_c.



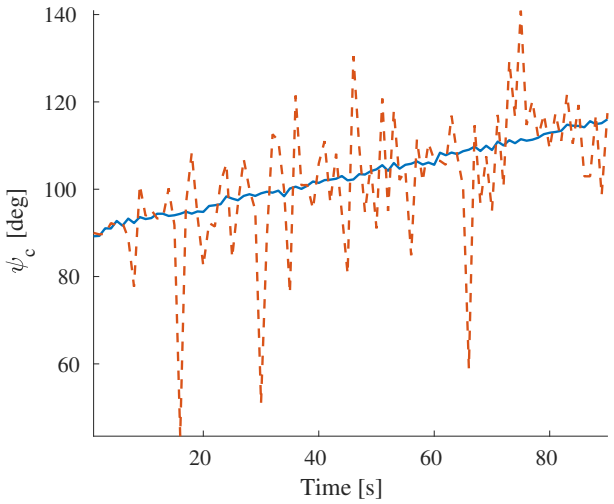
(b) East coordinate E_c.



(c) Velocity in North direction v_{N_c}.

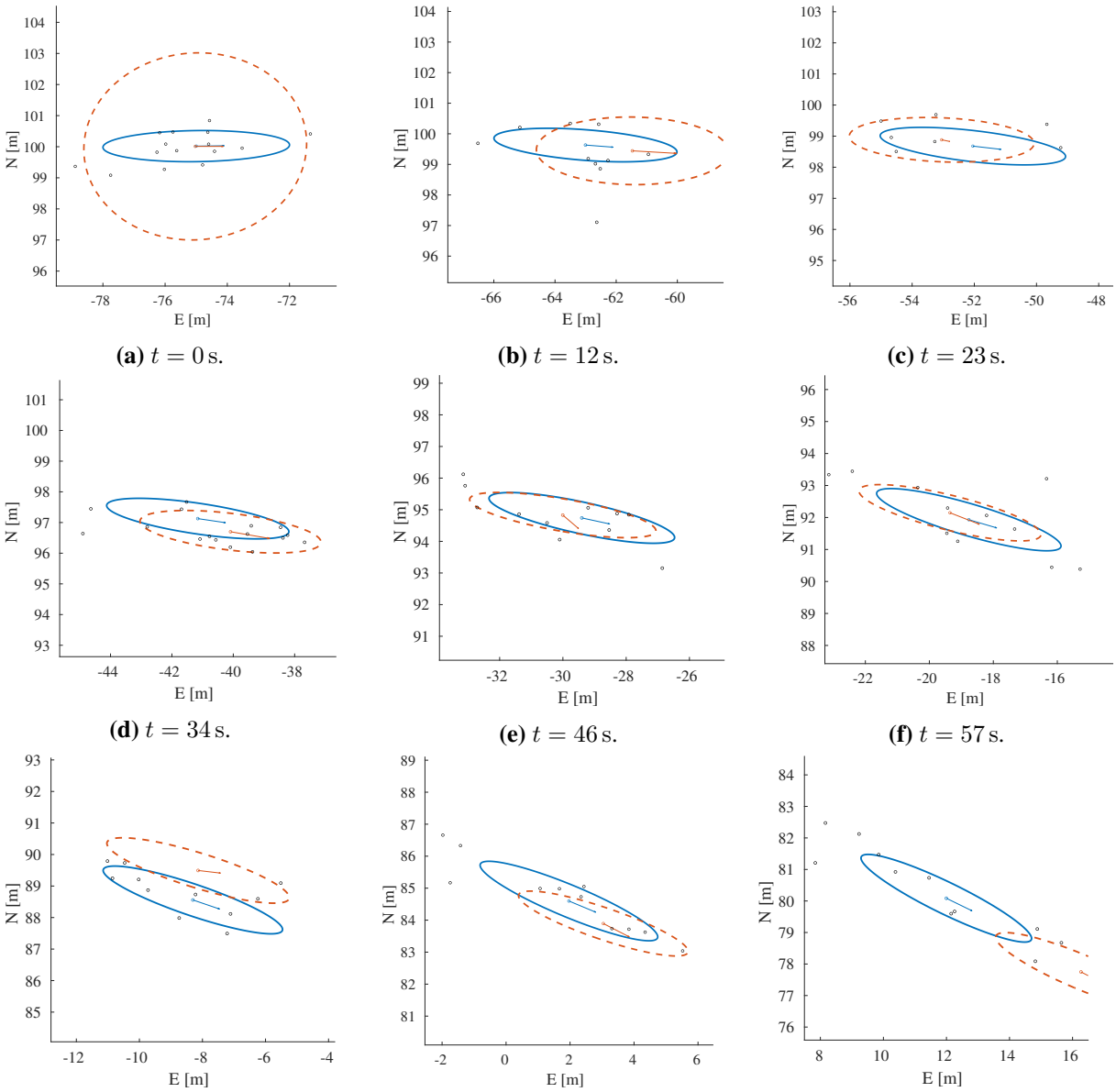


(d) Velocity in East direction v_{E_c}.



(e) Heading ψ_c.

Figure 7.13: Results for the kinematic states using the PMBM filter for MEOT and the GGIW model: Actual state values (blue) and corresponding state estimates (red dashed) of the first object generated at the West.

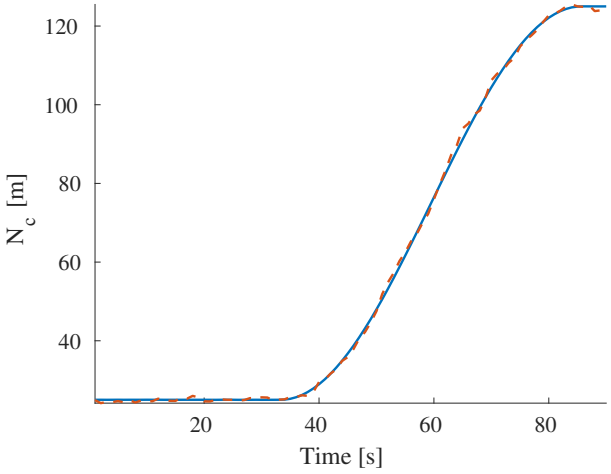


(a) $t = 0\,\mathrm{s}$. **(b)** $t = 12\,\mathrm{s}$. **(c)** $t = 23\,\mathrm{s}$.

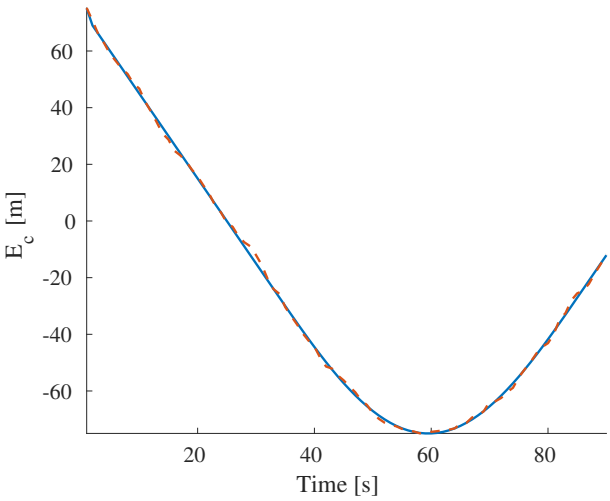
(d) $t = 34\,\mathrm{s}$. **(e)** $t = 46\,\mathrm{s}$. **(f)** $t = 57\,\mathrm{s}$.

(g) $t = 68\,\mathrm{s}$. **(h)** $t = 79\,\mathrm{s}$. **(i)** $t = 90\,\mathrm{s}$.

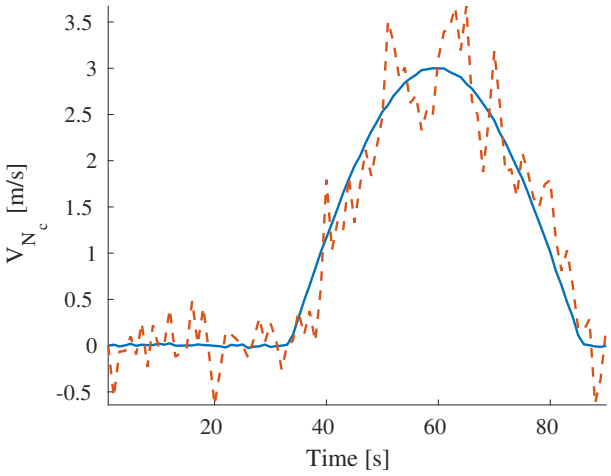
Figure 7.14: Time-lapse for the first object generated at the West using the PMBM filter for MEOT and the GGIW model: Actual extend (blue ellipsis), estimated extend (red dashed ellipsis), actual velocity vector (blue arrow), estimated velocity vector (red arrow) and radar measurements (black dots).



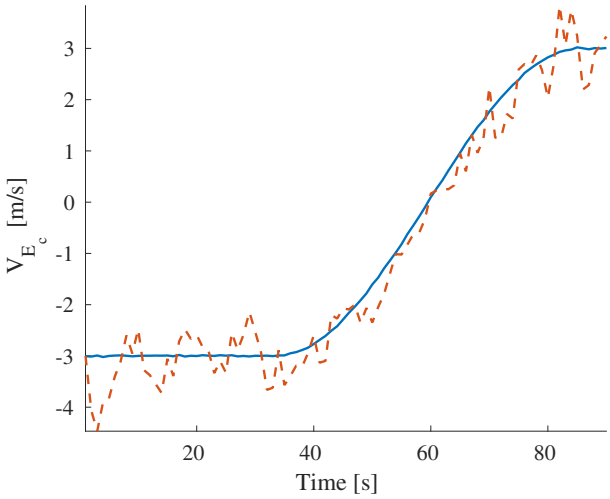
(a) North coordinate N_c.



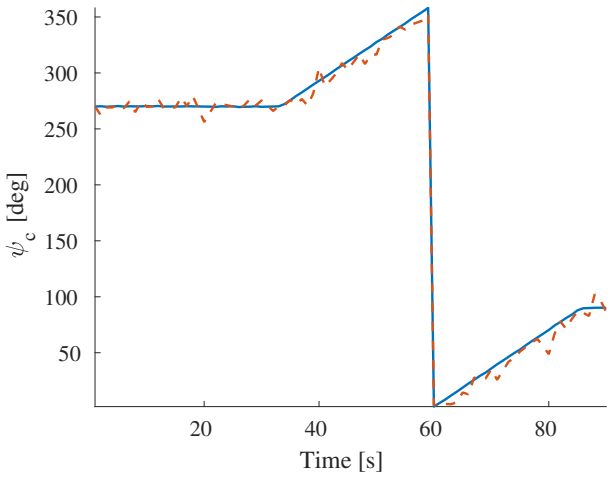
(b) East coordinate E_c.



(c) Velocity in North direction v_{N_c}.

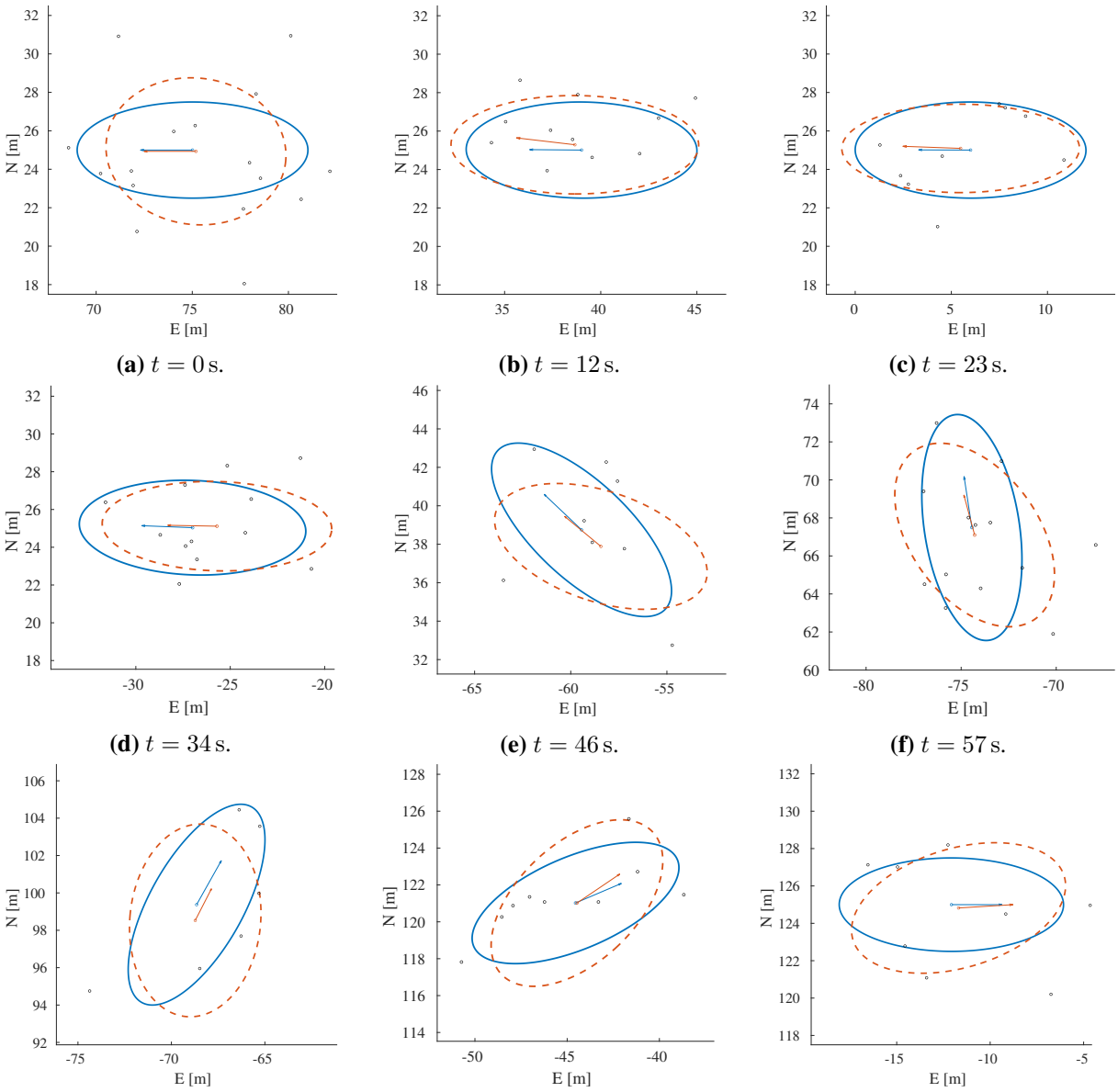


(d) Velocity in East direction v_{E_c}.



(e) Heading ψ_c.

Figure 7.15: Results for the kinematic states using the PMBM filter for MEOT and the GGIW model: Actual state values (blue) and corresponding state estimates (red dashed) of the first object generated at the East.



(a) $t = 0\,\mathrm{s}$. **(b)** $t = 12\,\mathrm{s}$. **(c)** $t = 23\,\mathrm{s}$.

(d) $t = 34\,\mathrm{s}$. **(e)** $t = 46\,\mathrm{s}$. **(f)** $t = 57\,\mathrm{s}$.

(g) $t = 68\,\mathrm{s}$. **(h)** $t = 79\,\mathrm{s}$. **(i)** $t = 90\,\mathrm{s}$.

Figure 7.16: Time-lapse for the first object generated at the East using the PMBM filter for MEOT and the GGIW model: Actual extend (blue ellipsis), estimated extend (red dashed ellipsis), actual velocity vector (blue arrow), estimated velocity vector (red arrow) and radar measurements (black dots).

7.2.2 Results using the GP and lidar model and discussion

The parameter values for the extends and the measurement model are the same as in Section 7.1.2, with the exception of the process noise strength σ_w used in the filter, whose value has to be considerably larger than the one in Section 7.1.2 due to the kinematic model discrepancies.

As done in Section 7.2.1, the process noise strength σ_w was increased to the round number $1\,\mathrm{m\,s^{-1}}$ in order to do not lose track of the objects.

Figure 7.17 illustrates the scenario for one simulation. Figure 7.18 and Figure 7.20 show the true and estimated kinematic state values for the objects generated at the west and east birth point, respectively, while Figure 7.19 and Figure 7.21 show the corresponding time-lapses for their extend trajectories and estimates.

Similarly to the results in Section 7.2.1, the kinematic state estimates are poor, specially the velocity and heading estimates, as shown in Figure 7.18 and Figure 7.20. Furthermore, we observe in Figure 7.19 and Figure 7.21 that the true extents lie considerably outside the estimated 99% confidence region. This is also a consequence of the discrepancy between the kinematic models used to simulate and to estimate.

As occurred in Section 7.2.1, the filter loses track of the object generated at the west birth point towards the end of the simulation as shown in Figure 7.19, while it manages to keep track of the other object. The reason the filter manages to keep track of the latter object is that this vessel moves in an almost straight line in the beginning of the simulation in accordance to the CV2 model without sideslip. This allows the filter to produce a good enough estimate of the object, which is robust enough to endure the half turn. However, in the case of the object generated at the west birth point, it turns from the start of the simulation, and the filter does not manage to handle both the initial estimation error and the kinematic model mismatch.

As occurred in Section 7.2.1, further tuning of the process noise strength could lead to a better performance for these particular simulations. However, this is not a sustainable strategy. The source of the poor estimates is endemic due to the kinematic model used in the filter.

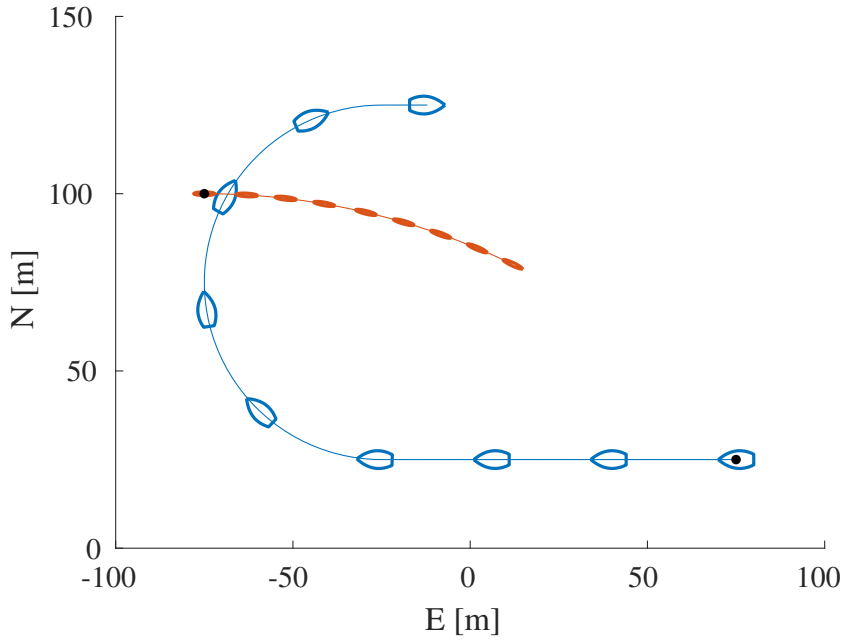
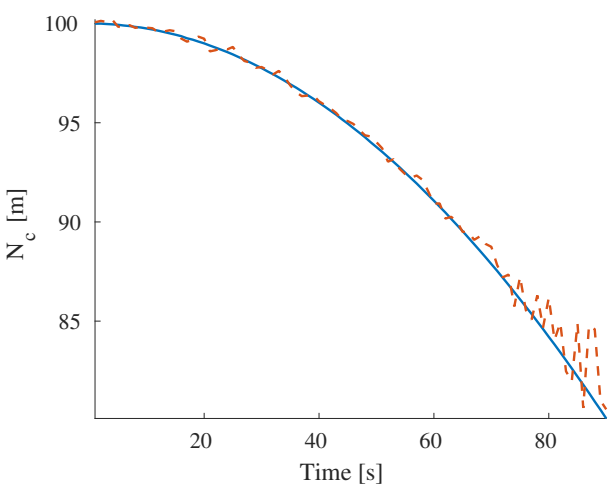
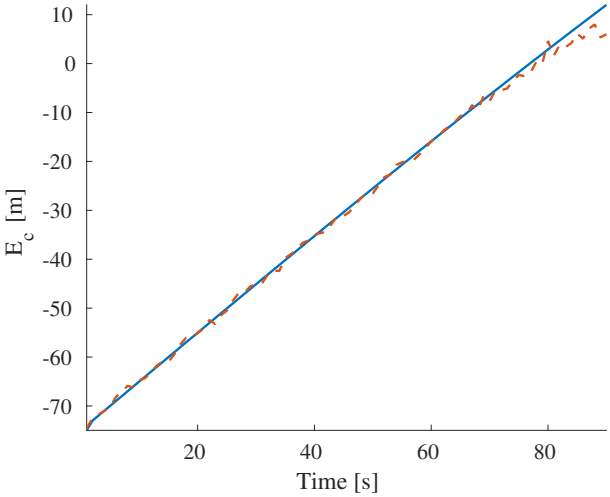


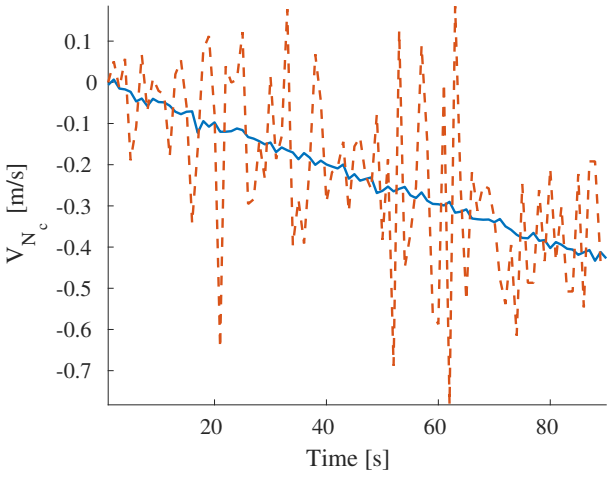
Figure 7.17: Coordinated turn simulation: Simulated trajectories for the different objects with a time-lapse of their extent positions. The black dots mark the points where objects are generated.



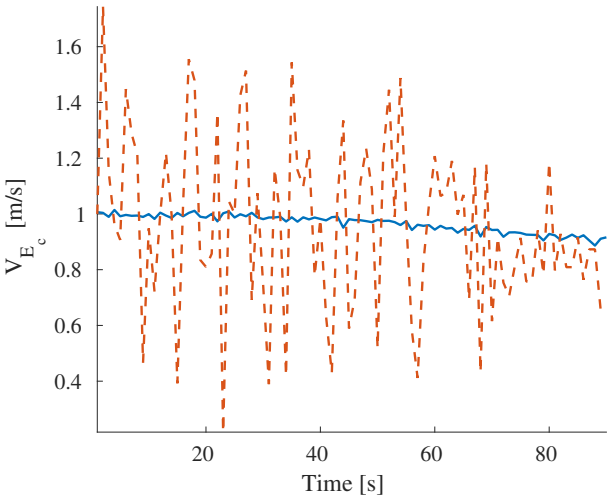
(a) North coordinate N_c.



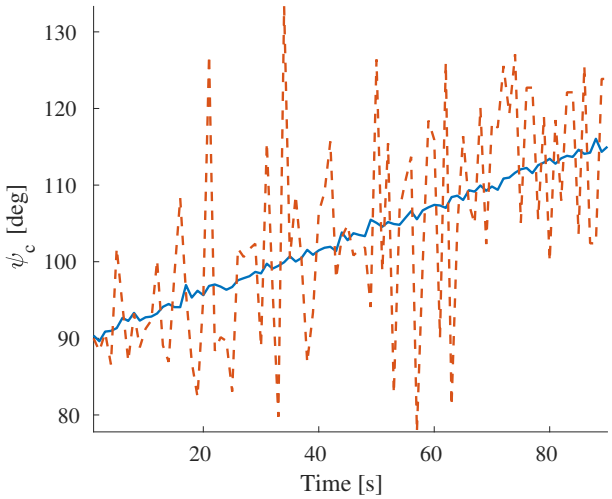
(b) East coordinate E_c.



(c) Velocity in North direction v_{N_c}.

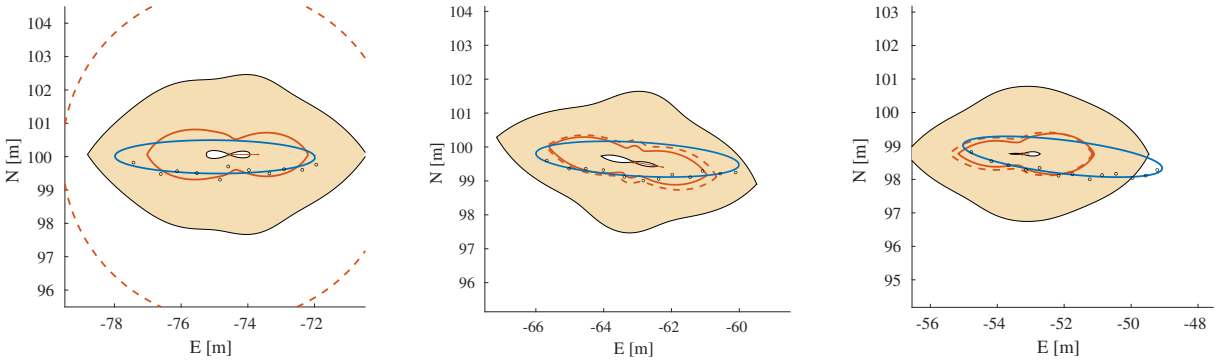


(d) Velocity in East direction v_{E_c}.



(e) Heading ψ_c.

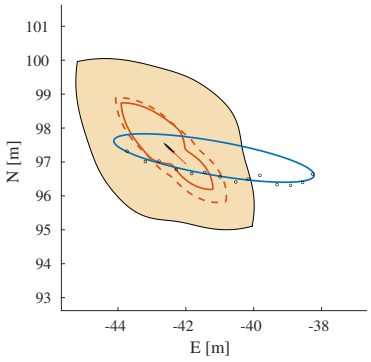
Figure 7.18: Results for the kinematic states using the PMBM filter for MEOT and the GP and lidar model: Actual state values (blue) and corresponding state estimates (red dashed) of the first object generated at the West.



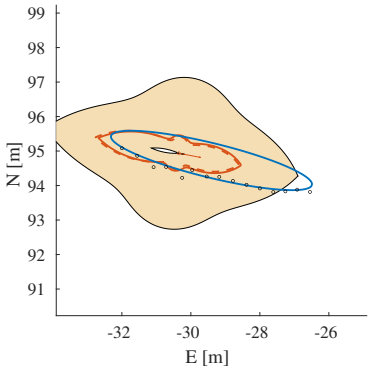
(a) $t = 0\,\mathrm{s}$.

(b) $t = 12\,\mathrm{s}$.

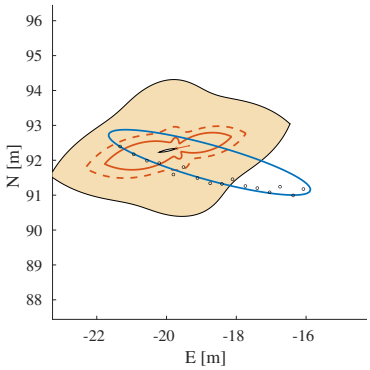
(c) $t = 23\,\mathrm{s}$.



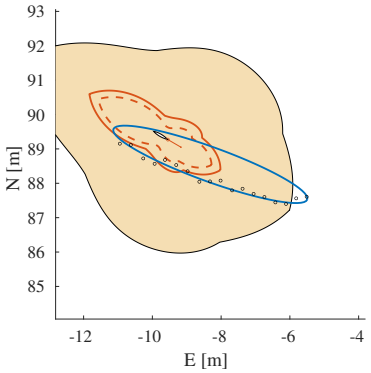
(d) $t = 34\,\mathrm{s}$.



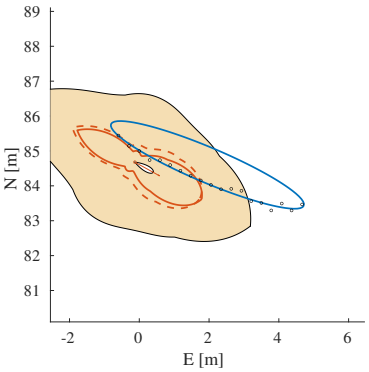
(e) $t = 46\,\mathrm{s}$.



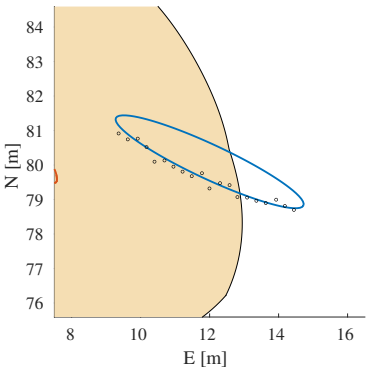
(f) $t = 57\,\mathrm{s}$.



(g) $t = 68\,\mathrm{s}$.

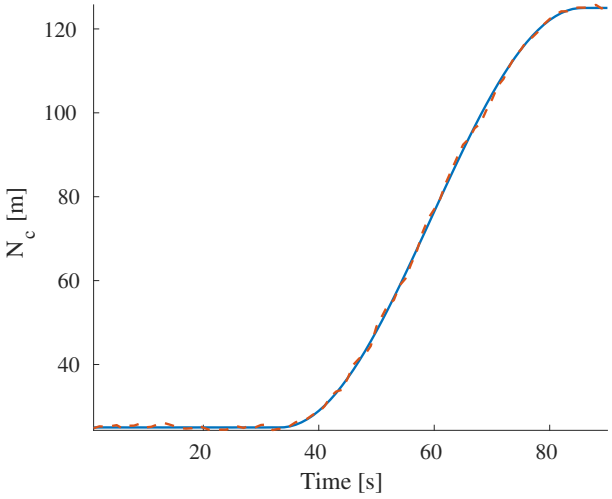


(h) $t = 79\,\mathrm{s}$.

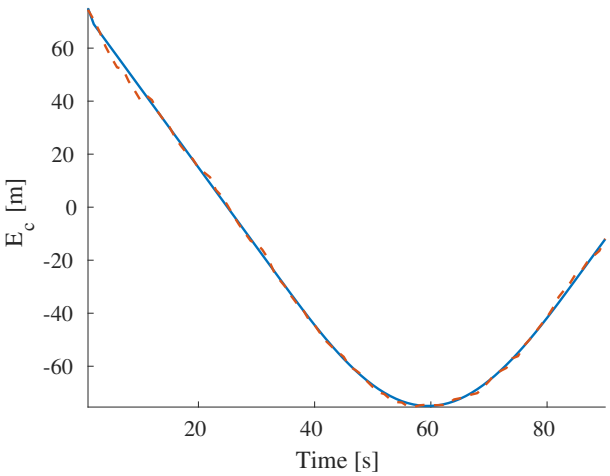


(i) $t = 90\,\mathrm{s}$.

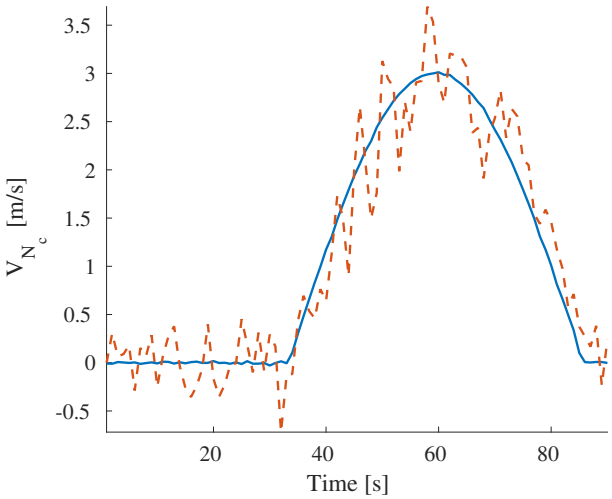
Figure 7.19: Time-lapse for the first object generated at the West using the PMBM filter for MEOT and the GP and lidar model: Actual extend (blue closed curve), predicted extend estimate (red dashed closed curve), updated extend estimate (red closed curve), 99% confidence region for the updated extend estimate (shadowed region), actual velocity vector (blue arrow), estimated velocity vector (red arrow) and lidar measurements (black dots).



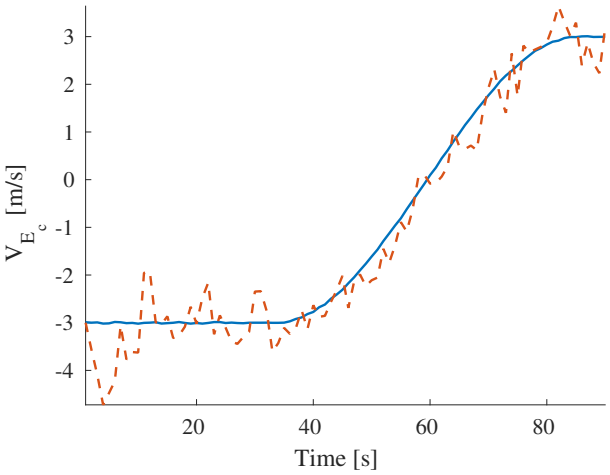
(a) North coordinate N_c.



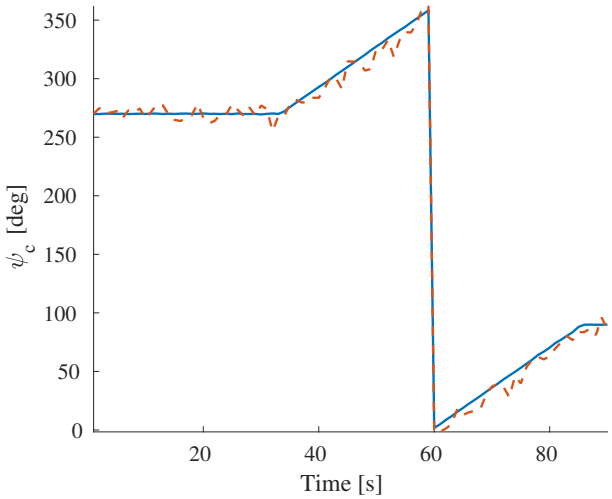
(b) East coordinate E_c.



(c) Velocity in North direction v_{N_c}.

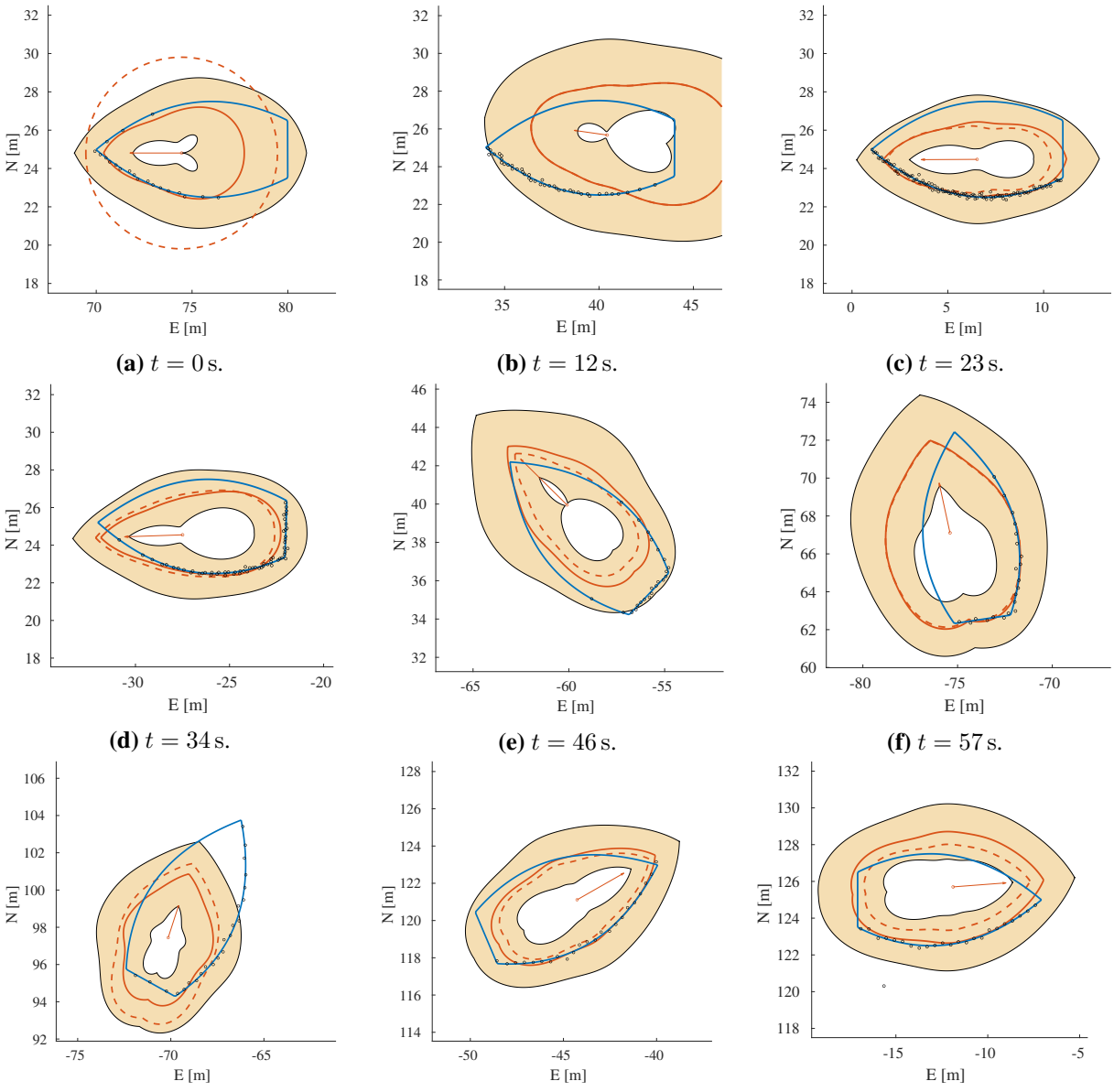


(d) Velocity in East direction v_{E_c}.



(e) Heading ψ_c.

Figure 7.20: Results for the kinematic states using the PMBM filter for MEOT and the GP and lidar model: Actual state values (blue) and corresponding state estimates (red dashed) of the first object generated at the East.



(a) $t = 0\,\mathrm{s}$. **(b)** $t = 12\,\mathrm{s}$. **(c)** $t = 23\,\mathrm{s}$.

(d) $t = 34\,\mathrm{s}$. **(e)** $t = 46\,\mathrm{s}$. **(f)** $t = 57\,\mathrm{s}$.

(g) $t = 68\,\mathrm{s}$. **(h)** $t = 79\,\mathrm{s}$. **(i)** $t = 90\,\mathrm{s}$.

Figure 7.21: Time-lapse for the first object generated at the East using the PMBM filter for MEOT and the GP and lidar model: Actual extend (blue closed curve), predicted extend estimate (red dashed closed curve), updated extend estimate (red closed curve), 99% confidence region for the updated extend estimate (shadowed region), actual velocity vector (blue arrow), estimated velocity vector (red arrow) and lidar measurements (black dots).

Chapter 8

Conclusions and further work

Gaussian Processes (GPes) provide a very versatile stochastic model for representing the extent of maritime vessels as shown in the examples of Section 5.1.3 and in the simulation results presented in Section 7.1.2 and Section 7.2.2.

Furthermore, the state-space model based on GPes (see Section 6.2) can be integrated with the Poisson multi-Bernoulli mixture (PMBM) filter for multiple extended object tracking (MEOT) derived in Chapter 4. The resulting MEOT method has potential as shown in Section 7.1.2. However, there is still room for improvement as noted in Section 7.2.2.

The fact that the process noise had to be increased in the coordinated turn simulations (see Section 7.2.2), so that the MEOT method would not lose track of the objects, may indicate that the non-linearities in the lidar measurement equation (6.37) have not been dealt with properly. A solution to this could be to use the extent model presented in [27], which is arguably more robust when associating measurements to the extent's boundary.

On the other side, the main difference between the results in Section 7.1.2 and Section 7.2.2 lie on how the kinematic states of the objects were simulated. In Section 7.1.2, they where simulated using constant velocity models, while in Section 7.2.2, the maritime vessels described a coordinated turn. Therefore, further work could revolve around implementing a PMBM filter with a hybrid state-transition model based on the master thesis of Tokle ([36]). In addition to the use of a CV2 model without sideslip (see Section 6.1.3), this hybrid kinematic state-transition model could also consist of a kinematic model suited for turns, such as a constant turn model[1], as well as a CV3 model for both velocity and turn rate, which could come in handy when the object has a very low velocity or when it describes sharp turns.

Another interesting route for further work and improvement is to fuse the lidar measurement model with radar or camera measurement models. By doing so, one could exploit the advantages of each individual sensor and cancel outs their deficiencies.

[1] A constant turn model is a kinematic model, where the object rotates with an almost constant angular velocity perturbed by white noise. This is similar to a CV model, where the object moves with an almost constant velocity perturbed by white noise.

Appendix A

Results on multivariate Gaussian distributions

Theorem A.1. *(The conditional multivariate normal distribution) Let* $\mathbf{x}$ *and* $\mathbf{y}$ *be two stochastic variables that have a joint multivariate normal distribution, i.e.,*

$$\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\boldsymbol{\mu}_{\mathbf{x}}\\ \boldsymbol{\mu}_{\mathbf{y}}\end{bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma}_{\mathbf{xx}} & \boldsymbol{\Sigma}_{\mathbf{xy}}\\ \boldsymbol{\Sigma}_{\mathbf{yx}} & \boldsymbol{\Sigma}_{\mathbf{yy}}\end{bmatrix}\right), \tag{A.1}$$

where $\boldsymbol{\Sigma}_{\mathbf{yx}} = \boldsymbol{\Sigma}_{\mathbf{xy}}^T$. *Then the conditional distribution of* $\mathbf{y}$ *given* $\mathbf{x}$ *is*

$$\mathbf{y}|\mathbf{x} \sim \mathcal{N}\left(\boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{y}|\mathbf{x}}\right), \tag{A.2}$$

where

$$\boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}} = \boldsymbol{\mu}_{\mathbf{y}} + \boldsymbol{\Sigma}_{\mathbf{yx}}\boldsymbol{\Sigma}_{\mathbf{xx}}^{-1}\left[\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}}\right] \tag{A.3a}$$

$$\boldsymbol{\Sigma}_{\mathbf{y}|\mathbf{x}} = \boldsymbol{\Sigma}_{\mathbf{yy}} - \boldsymbol{\Sigma}_{\mathbf{yx}}\boldsymbol{\Sigma}_{\mathbf{xx}}^{-1}\boldsymbol{\Sigma}_{\mathbf{xy}}. \tag{A.3b}$$

Proof. The bilinear form in the exponent of the joint probability density function $p(\mathbf{x}, \mathbf{y})$ can be decomposed as

$$\begin{bmatrix}\mathbf{x}^T - \boldsymbol{\mu}_{\mathbf{x}}^T & \mathbf{y}^T - \boldsymbol{\mu}_{\mathbf{y}}^T\end{bmatrix}\begin{bmatrix}\boldsymbol{\Sigma}_{\mathbf{xx}} & \boldsymbol{\Sigma}_{\mathbf{xy}}\\ \boldsymbol{\Sigma}_{\mathbf{yx}} & \boldsymbol{\Sigma}_{\mathbf{yy}},\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}}\\ \mathbf{y} - \boldsymbol{\mu}_{\mathbf{y}}\end{bmatrix} \tag{A.4a}$$

$$= \left[\mathbf{y} - \boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}}\right]^T\boldsymbol{\Sigma}_{\mathbf{y}|\mathbf{x}}^{-1}\left[\mathbf{y} - \boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}}\right] + \left[\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}}\right]^T\boldsymbol{\Sigma}_{\mathbf{xx}}^{-1}\left[\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}}\right]. \tag{A.4b}$$

Hence, for a fixed vector $\mathbf{x}$, the conditional probability density function $p(\mathbf{y}|\mathbf{x})$ is

$$p(\mathbf{y}|\mathbf{x}) = \frac{p(\mathbf{x}, \mathbf{y})}{\int p(\mathbf{x}, \mathbf{y})\, d\mathbf{y}} \tag{A.5a}$$

$$= \frac{e^{\left(-\frac{1}{2}\left[\mathbf{y}-\boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}}\right]^T\boldsymbol{\Sigma}_{\mathbf{y}|\mathbf{x}}^{-1}\left[\mathbf{y}-\boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}}\right]\right)}e^{\left(-\frac{1}{2}\left[\mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}}\right]^T\boldsymbol{\Sigma}_{\mathbf{xx}}^{-1}\left[\mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}}\right]\right)}}{\int e^{\left(-\frac{1}{2}\left[\mathbf{y}-\boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}}\right]^T\boldsymbol{\Sigma}_{\mathbf{y}|\mathbf{x}}^{-1}\left[\mathbf{y}-\boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}}\right]\right)}e^{\left(-\frac{1}{2}\left[\mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}}\right]^T\boldsymbol{\Sigma}_{\mathbf{xx}}^{-1}\left[\mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}}\right]\right)}\, d\mathbf{y}} \tag{A.5b}$$

$$= \frac{e^{\left(-\frac{1}{2}\left[\mathbf{y}-\boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}}\right]^T\boldsymbol{\Sigma}_{\mathbf{y}|\mathbf{x}}^{-1}\left[\mathbf{y}-\boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}}\right]\right)}e^{\left(-\frac{1}{2}\left[\mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}}\right]^T\boldsymbol{\Sigma}_{\mathbf{xx}}^{-1}\left[\mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}}\right]\right)}}{(2\pi)^{\frac{n}{2}}\left|\Sigma_{\mathbf{y}|\mathbf{x}}\right|^{\frac{1}{2}}e^{\left(-\frac{1}{2}\left[\mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}}\right]^T\boldsymbol{\Sigma}_{\mathbf{xx}}^{-1}\left[\mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}}\right]\right)}} \tag{A.5c}$$

$$= \frac{e^{\left(-\frac{1}{2}\left[\mathbf{y}-\boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}}\right]^T\boldsymbol{\Sigma}_{\mathbf{y}|\mathbf{x}}^{-1}\left[\mathbf{y}-\boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}}\right]\right)}}{(2\pi)^{\frac{n}{2}}\left|\Sigma_{\mathbf{y}|\mathbf{x}}\right|^{\frac{1}{2}}} \sim \mathcal{N}\left(\boldsymbol{\mu}_{\mathbf{y}|\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{y}|\mathbf{x}}\right). \tag{A.5d}$$

□

Theorem A.2. *Let* $\mathbf{x}$ *and* $\mathbf{z}_1, \ldots, \mathbf{z}_n$ *be stochastic variables distributed as*

$$\mathbf{x} \sim \mathcal{N}\left(\boldsymbol{\mu}_{\mathbf{x}}, \mathbf{P}\right) \tag{A.6a}$$

$$\mathbf{z}_i|\mathbf{x} \sim \mathcal{N}\left(\mathbf{A}_i\mathbf{x}, \mathbf{Q}_i\right). \tag{A.6b}$$

Then the product of the probability density functions of the distributions in (A.6a) is

$$\mathcal{N}\left(\mathbf{x}, \boldsymbol{\mu}_{\mathbf{x}}, \mathbf{P}\right) \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{z}, \mathbf{A}_i\mathbf{x}, \mathbf{Q}_i\right) = \mathcal{N}\left(\begin{bmatrix} \mathbf{x} \\ \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_n \end{bmatrix}, \boldsymbol{\mu}, \boldsymbol{\Sigma}\right), \tag{A.7}$$

where

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_{\mathbf{x}} \\ \mathbf{A}_1\boldsymbol{\mu}_{\mathbf{x}} \\ \vdots \\ \mathbf{A}_n\boldsymbol{\mu}_{\mathbf{x}} \end{bmatrix} \tag{A.8a}$$

$$\boldsymbol{\Sigma} = \begin{bmatrix} \mathbf{P} & \mathbf{P}\mathbf{A}_1^T & \mathbf{P}\mathbf{A}_2^T & \cdots & \cdots & \mathbf{P}\mathbf{A}_n^T \\ \mathbf{A}_1\mathbf{P} & \mathbf{A}_1\mathbf{P}\mathbf{A}_1^T + \mathbf{Q}_1 & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{A}_2\mathbf{P} & \mathbf{0} & \mathbf{A}_2\mathbf{P}\mathbf{A}_2^T + \mathbf{Q}_2 & \mathbf{0} & \ddots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ \mathbf{A}_n\mathbf{P} & \mathbf{0} & \cdots & \cdots & \mathbf{0} & \mathbf{A}_n\mathbf{P}\mathbf{A}_n^T + \mathbf{Q}_n \end{bmatrix}. \tag{A.8b}$$

Proof. We prove first that the bilinear forms in the exponents for both sides in (A.7) are equal.

The bilinear form for the left-hand side of (A.7) is

$$(\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}})^T\mathbf{P}^{-1}(\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}}) + (\mathbf{z}_1 - \mathbf{A}_1\mathbf{x})^T\mathbf{R}_1^{-1}(\mathbf{z}_1 - \mathbf{A}_1\mathbf{x})^T + \cdots + (\mathbf{z}_n - \mathbf{A}_n\mathbf{x})^T\mathbf{R}_n^{-1}(\mathbf{z}_n - \mathbf{A}_n\mathbf{x})^T \tag{A.9a}$$

$$= \begin{bmatrix} \mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}} \\ \mathbf{z}_1 - \mathbf{A}_1\mathbf{x} \\ \vdots \\ \mathbf{z}_n - \mathbf{A}_n\mathbf{x} \end{bmatrix}^T \begin{bmatrix} \mathbf{P}^{-1} & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_1^{-1} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \cdots & \mathbf{0} & \mathbf{Q}_n^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}} \\ \mathbf{z}_1 - \mathbf{A}_1\mathbf{x} \\ \vdots \\ \mathbf{z}_n - \mathbf{A}_n\mathbf{x} \end{bmatrix}. \tag{A.9b}$$

By inserting the identity

$$\begin{bmatrix} \mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}} \\ \mathbf{z}_1 - \mathbf{A}_1\mathbf{x} \\ \vdots \\ \mathbf{z}_n - \mathbf{A}_n\mathbf{x} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ -\mathbf{A}_1 & \mathbf{I} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ -\mathbf{A}_n & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}} \\ \mathbf{z}_1 - \mathbf{A}_1\boldsymbol{\mu}_{\mathbf{x}} \\ \vdots \\ \mathbf{z}_n - \mathbf{A}_n\boldsymbol{\mu}_{\mathbf{x}} \end{bmatrix} \tag{A.10}$$

into (A.9b), we conclude that

$$\begin{bmatrix} \mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}} \\ \mathbf{z}_1-\mathbf{A}_1\mathbf{x} \\ \vdots \\ \mathbf{z}_n-\mathbf{A}_n\mathbf{x} \end{bmatrix}^T \begin{bmatrix} \mathbf{P}^{-1} & \mathbf{0} & \dots & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_1^{-1} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \dots & \dots & \mathbf{0} & \mathbf{Q}_n^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}} \\ \mathbf{z}_1-\mathbf{A}_1\mathbf{x} \\ \vdots \\ \mathbf{z}_n-\mathbf{A}_n\mathbf{x} \end{bmatrix} \tag{A.11a}$$

$$= \begin{bmatrix} \mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}} \\ \mathbf{z}_1-\mathbf{A}_1\boldsymbol{\mu}_{\mathbf{x}} \\ \vdots \\ \mathbf{z}_n-\mathbf{A}_n\boldsymbol{\mu}_{\mathbf{x}} \end{bmatrix}^T \begin{bmatrix} \mathbf{I} & -\mathbf{A}_1^T & \dots & \dots & -\mathbf{A}_n^T \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \dots & \dots & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{P}^{-1} & \mathbf{0} & \dots & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_1^{-1} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \dots & \dots & \mathbf{0} & \mathbf{Q}_n^{-1} \end{bmatrix} \times \dots \tag{A.11b}$$

$$\dots \begin{bmatrix} \mathbf{I} & \mathbf{0} & \dots & \dots & \mathbf{0} \\ -\mathbf{A}_1 & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ -\mathbf{A}_n & \mathbf{0} & \dots & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}} \\ \mathbf{z}_1-\mathbf{A}_1\boldsymbol{\mu}_{\mathbf{x}} \\ \vdots \\ \mathbf{z}_n-\mathbf{A}_n\boldsymbol{\mu}_{\mathbf{x}} \end{bmatrix} \tag{A.11c}$$

$$= \begin{bmatrix} \mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}} \\ \mathbf{z}_1-\mathbf{A}_1\boldsymbol{\mu}_{\mathbf{x}} \\ \vdots \\ \mathbf{z}_n-\mathbf{A}_n\boldsymbol{\mu}_{\mathbf{x}} \end{bmatrix}^T \left(\begin{bmatrix} \mathbf{I} & \mathbf{0} & \dots & \dots & \mathbf{0} \\ \mathbf{A}_1 & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ \mathbf{A}_n & \mathbf{0} & \dots & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} P & \mathbf{0} & \dots & \dots & \mathbf{0} \\ \mathbf{0} & Q_1 & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \dots & \dots & \mathbf{0} & Q_n \end{bmatrix} \times \dots \tag{A.11d}$$

$$\dots \begin{bmatrix} \mathbf{I} & \mathbf{A}_1^T & \dots & \dots & \mathbf{A}_n^T \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \dots & \dots & \mathbf{0} & \mathbf{I} \end{bmatrix} \Bigg)^{-1} \begin{bmatrix} \mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}} \\ \mathbf{z}_1-\mathbf{A}_1\boldsymbol{\mu}_{\mathbf{x}} \\ \vdots \\ \mathbf{z}_n-\mathbf{A}_n\boldsymbol{\mu}_{\mathbf{x}} \end{bmatrix} \tag{A.11e}$$

$$= \begin{bmatrix} \mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}} \\ \mathbf{z}_1-\mathbf{A}_1\boldsymbol{\mu}_{\mathbf{x}} \\ \vdots \\ \mathbf{z}_n-\mathbf{A}_n\boldsymbol{\mu}_{\mathbf{x}} \end{bmatrix}^T \begin{bmatrix} \mathbf{P} & \mathbf{P}\mathbf{A}_1^T & \mathbf{P}\mathbf{A}_2^T & \dots & \dots & \mathbf{P}\mathbf{A}_n^T \\ \mathbf{A}_1\mathbf{P} & \mathbf{A}_1\mathbf{P}\mathbf{A}_1^T+\mathbf{Q}_1 & \mathbf{0} & \dots & \dots & \mathbf{0} \\ \mathbf{A}_2\mathbf{P} & \mathbf{0} & \mathbf{A}_2\mathbf{P}\mathbf{A}_2^T+\mathbf{Q}_2 & \mathbf{0} & \ddots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \mathbf{0} \\ \mathbf{A}_n\mathbf{P} & \mathbf{0} & \dots & \dots & \mathbf{0} & \mathbf{A}_n\mathbf{P}\mathbf{A}_n^T+\mathbf{Q}_n. \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{x}-\boldsymbol{\mu}_{\mathbf{x}} \\ \mathbf{z}_1-\mathbf{A}_1\boldsymbol{\mu}_{\mathbf{x}} \\ \vdots \\ \mathbf{z}_n-\mathbf{A}_n\boldsymbol{\mu}_{\mathbf{x}} \end{bmatrix} \tag{A.11f}$$

$$= \left(\begin{bmatrix} \mathbf{x} \\ \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_n \end{bmatrix} - \boldsymbol{\mu} \right)^T \boldsymbol{\Sigma}^{-1} \left(\begin{bmatrix} \mathbf{x} \\ \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_n \end{bmatrix} - \boldsymbol{\mu} \right). \tag{A.11g}$$

Hence, the bilinear forms in the exponents for both sides in (A.7) are equal. This implies

that

$$\mathcal{N}\left(\mathbf{x}, \boldsymbol{\mu}_{\mathbf{x}}, \mathbf{P}\right) \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{z}, \mathbf{A}_i \mathbf{x}, \mathbf{Q}_i\right) = \kappa \mathcal{N}\left(\begin{bmatrix} \mathbf{x} \\ \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_n \end{bmatrix}, \boldsymbol{\mu}, \boldsymbol{\Sigma} \right) \tag{A.12}$$

for some constant κ. However, since probability density functions integrate up to 1, integration of (A.7) with respect to $\mathbf{x}$ and all $\mathbf{z}_i$ gives that $\kappa = 1$, and the final result follows. □

Theorem A.3. *(Gaussian mixture) Let X be a random variable whose probability density function is the weighted average of a finite number of multivariate Gaussian distributions, i.e.,*

$$f_X(\mathbf{x}) = \sum_{i=1}^{n} w_n \mathcal{N}\left(\mathbf{x}, \mathbf{x}_i, \mathbf{P}_i\right) , \tag{A.13}$$

where the weights $w_i \geq 0$ are normalized, i.e., $\sum_{i=1}^{n} w_i = 1$, and the multivariate normal distributions $\mathcal{N}\left(\mathbf{x}_i, \mathbf{P}_i\right)$ are independent of each other.

Then the mean value and covariance matrix of X are respectively

$$\mathbb{E}[X] = \sum_{i=1}^{n} w_i \mathbf{x}_i \tag{A.14a}$$

$$\mathrm{Cov}[X] = \sum_{i=1}^{n} w_i \mathbf{P}_i + \sum_{i=1}^{n} w_i \mathbf{x}_i \mathbf{x}_i^T - \mathbb{E}[X]\mathbb{E}[X]^T. \tag{A.14b}$$

Proof. The proof is an exercise in mean and covariance calculations, and is left to the reader. For a proof see page 56 in [6].

□

Notation, abbreviations and symbols

Lower case letters (e.g. a, ψ, f) denote scalars, lower case bold letters (e.g. $\mathbf{x}$, $\mathbf{z}$) denote vectors and upper case bold letters (e.g. $\mathbf{F}$, $\mathbf{H}$, $\mathbf{X}$) denote matrices. Furthermore, random finite sets (RFSs) are denoted by calligraphic letters (e.g. $\mathcal{A}$, $\mathcal{X}$, $\mathcal{Z}$) and index sets are denoted by blackboard letters (e.g. $\mathbb{I}$, $\mathbb{M}$).

Abbreviations

AIS Automatic identification system.

ASV Autonomous surface vehicles

CV Constant velocity.

EKF Extended Kalman filter.

EOT Extended Object Tracking.

FISST Finite Set Statistics.

GGIW Gamma Gaussian inverse Wishart.

GNSS Global navigation satellite system.

GP Gaussian Process.

IMU Inertial measurement unit.

KF Kalman filter.

LIDAR Light detection and ranging.

MB Multi-Bernoulli.

MBM Multi-Bernoulli mixture.

MEOT Multiple Extended Object Tracking.

MOT Multiple Object Tracking.

NTNU Norwegian University of Science and Technology.

OT Object Tracking.

PGFL Probability generating functional.

PMBM Poisson multi-Bernoulli mixture.

PPP Point Poisson Process.

RCC Remote control center.

RFS Random Finite Set.

Gaussian Processes

$r(\theta)$ Radius function that parameterizes the extent boundary.

$f \sim \mathcal{GP}(m, k)$ f is a Gaussian process with mean function m and covariance function k.

$m(\cdot)$ Mean function of a Gaussian process.

$k(\cdot, \cdot)$ Covariance function of a Gaussian process.

$\tilde{k}_{2\pi}(\theta, \theta')$ Pure exponential part of $k_{2\pi}$.

$\tilde{k}_{2\pi,a}(\theta, \theta')$ Pure exponential part of $k_{2\pi,a}$.

σ_f Variance of the signal amplitude. A hyperparameter.

σ_b Variance of the bias. A hyperparameter.

σ_n Variance of the noise. A hyperparameter.

$\mathbf{M}(\mathbf{i})$ Associated mean vector to index vector $\mathbf{i}$.

$\mathbf{K}(\mathbf{i}, \mathbf{j})$ Associated covariance matrix to index vectors $\mathbf{i}$ and $\mathbf{j}$.

l Length-scale. A hyperparameter

$\delta(\theta, \theta')$ Dirac delta.

$k_e(\theta, \theta')$ Squared exponential covariance function.

$\mathrm{ssa}(\theta)$ Smallest signed angle of θ.

$k_{2\pi}(\theta, \theta')$ Covariance function for 2π-periodic functions.

$k_{2\pi,a}(\theta, \theta')$ Covariance function for 2π-periodic and even functions.

$\tilde{k}_e(\theta, \theta')$ Pure exponential part of k_e.

General

$\mathbb{E}[\cdot]$ Expected value or mean value operator.

$Cov[\cdot]$ Covariance matrix operator.

$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ Multivariate normal distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$.

$\mathcal{N}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ Probability density function of $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with variable $\mathbf{x}$.

$\mathbf{I}_n$ n-by-n identity matrix.

$\otimes$ Kronecker product of matrices.

$(\operatorname{mod} \cdot)$ Modulo congruence relation.

$\operatorname{atan2}(\cdot, \cdot)$ Four quadrant arctangent.

$\angle \mathbf{x}$ Angle that gives the direction of the vector $\mathbf{x}$.

State-space models

$\mathbf{x}$ State vector of object.

$\mathbf{x}_c$ Kinematic state vector.

$\mathbf{x}_e$ Extent state vector.

c Center of movement.

$\mathbf{r}_c$ Position of center of movement, i.e., of object.

N_c North coordinate of the center of movement.

E_c East coordinates of the center of movement.

v_{N_c} Velocity of the object in the North direction.

v_{E_c} Velocity of the object in the East direction.

ψ_c Heading angle.

r_c Turn rate.

$(\hat{N}, \hat{E})$ World reference axes.

$(\hat{x}, \hat{y})$ Axes of local reference frame attached to extent, i.e., body frame.

r_i Lidar range measurement.

φ_i Lidar azimuth angle.

σ_r Measurement noise strength for lidar or radar measurements.

σ_w Process noise strength for CV model.

T Constant time step.

$\mathbf{e}(\cdot)$ Unit vector function.

$\mathbf{X}$ Random matrix.

Poisson multi-Bernoulli mixture filter

$D^b(\mathbf{x})$ Birth intensity.

$p^s(\mathbf{x})$ Survival probability.

$D^c(\mathbf{x})$ Clutter (or false alarm) intensity.

$p^d(\mathbf{x})$ Detection probability.

$\lambda^m(\mathbf{x})$ Poisson rate of object-generated measurements.

$q^d(\mathbf{x})$ The effective probability of missed detection.

$l_Z(\mathbf{x})$ The likelihood of a measurement set Z given a state vector $\mathbf{x}$

A Data association hypothesis.

C Cell in a data association hypothesis.

$l_C(\mathbf{x})$ The likelihood of a measurement set Z_C given a state vector $\mathbf{x}$

w^j Probability of a data association hypothesis A.

D^u The PPP intensity of undetected targets.

$\mathcal{X}^u$ RFS for the undetected objects.

$\mathcal{X}^t$ RFS for the tracked objects.

$\mathcal{X}$ RFS for the existing objects.

$\mathcal{X}_k$ Updated RFS for $\mathcal{X}$.

$\mathcal{A}^j$ Set of all possible data associations for a global hypothesis with index j.

Random Finite Sets

$\mathcal{X}$ Random finite set (RFS).

X Realization of a RFS.

$\mathcal{P}_F(\mathbb{S})$ Set of all finite subsets of $\mathbb{S}$.

$\mathbb{S}$ Base space of a RFS. Usually $\mathbb{S} = \mathbb{R}^d$

$f_{\mathcal{X}}(X)$ Set density function.

$\Pr(A)$ Probability of event A.

σ Permutation.

S_n Set of all permutations on $\{1, \ldots, n\}$.

$\biguplus$ Union, where the sets being united are disjoint 2-by-2.

$h(\cdot)$ Test function.

$< f, g >$ Scalar product of two functions, i.e., integral of their product.

$F_{\mathcal{X}}$ Probability generating functional (PGFL) of the RFS $\mathcal{X}$.

h^X Product of $h(\mathbf{x})$ for all elements $\mathbf{x} \in X$. If X is empty, then it is equal to 1.

$\int f(X)\delta X$ Set integral of set function f..

$\frac{\delta F}{\delta \mathbf{x}}$ Functional derivative of functional F with respect to the vector $\mathbf{x}$.

$\frac{\delta F}{\delta X}$ Functional derivative of functional F with respect to the set X.

$\delta_{\mathbf{x}}(\cdot)$ Delta function concentrated on point $\mathbf{x}$.

Bibliography

[1] Autoferry - NTNU. https://www.ntnu.edu/autoferry. Last accessed: 09.08.20.

[2] Autosea - NTNU. https://www.ntnu.edu/autosea. Last accessed: 09.08.20.

[3] Autosit - NTNU. https://www.ntnu.edu/autosit. Last accessed: 09.08.20.

[4] The ReVolt DNV-GL. https://www.dnvgl.com/technology-innovation/revolt/index.html. Last accessed: 09.08.20.

[5] Velodyne Lidar Puck™. https://velodynelidar.com/vlp-16.html. Last accessed: 09.08.20.

[6] Yaakov Bar-Shalom, X Rong Li, and Thiagalingam Kirubarajan. *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.

[7] Marcus Baum and Uwe D Hanebeck. Extended object tracking with random hypersurface models. *IEEE Transactions on Aerospace and Electronic systems*, 50(1):149–159, 2014.

[8] Edmund Brekke. Sensor fusion and collision avoidance for autonomous surface vehicles. https://haugesundkonferansen.no/content/uploads/2018/09/Edmund-Forland-Brekke.pdf. Last accessed: 09.08.20.

[9] Donald L Cohn. *Measure theory*. Springer, 2013.

[10] Emil Aall Dahle. Selvkjørende AI-skip – en dårlig ide? https://www.tu.no/artikler/selvkjorende-ai-skip-en-darlig-ide/491460, April 2020. Last accessed: 09.08.20.

[11] OE Drummond, SS Blackman, and KC Hell. Multiple sensor tracking of clusters and extended objects. In *Technical Proceedings*, 1988.

[12] Michael Feldmann, Dietrich Franken, and Wolfgang Koch. Tracking of extended objects and group targets using random matrices. *IEEE Transactions on Signal Processing*, 59(4):1409–1420, 2010.

[13] Thor I Fossen. Marine control system-guidance, navigation and control of ships, rigs and underwater vehicles. *Marine Cybemetics*, 2002.

[14] Karl Granström, Maryam Fatemi, and Lennart Svensson. Poisson multi-bernoulli mixture conjugate prior for multiple extended target filtering. *IEEE Transactions on Aerospace and Electronic Systems*, 56(1):208–225, 2019.

[15] Karl Granström, Christian Lundquist, and Umut Orguner. Tracking rectangular and elliptical extended targets using laser measurements. In *14th International Conference on Information Fusion*, pages 1–8. IEEE, 2011.

[16] Karl Granström and Umut Orguner. New prediction for extended targets with random matrices. *IEEE Transactions on Aerospace and Electronic Systems*, 50(2):1577–1589, 2014.

[17] Karl Granström, Lennart Svensson, Stephan Reuter, Yuxuan Xia, and Maryam Fatemi. Likelihood-based data association for extended object tracking using sampling methods. *IEEE Transactions on intelligent vehicles*, 3(1):30–45, 2017.

[18] Karl Granström, Lennart Svensson, Yuxuan Xia, Jason Williams, and Ángel F García-Femández. Poisson multi-bernoulli mixture trackers: Continuity through random finite sets of trajectories. In *2018 21st International Conference on Information Fusion (FUSION)*, pages 1–5. IEEE, 2018.

[19] Karl Granström, Marcus Baum, and Stephan Reuter. Extended object tracking: Introduction, overview and applications. *arXiv preprint arXiv:1604.00970*, 2016.

[20] Tobias Hirscher, Alexander Scheel, Stephan Reuter, and Klaus Dietmayer. Multiple extended object tracking using gaussian processes. In *2016 19th International Conference on Information Fusion (FUSION)*, pages 868–875. IEEE, 2016.

[21] Johann Wolfgang Koch. Bayesian approach to extended object and cluster tracking using random matrices. *IEEE Transactions on Aerospace and Electronic Systems*, 44(3):1042–1059, 2008.

[22] Andrei N. Kolmogorov. Interpolation und Extrapolation von stationären zufälligen Folgen. *Bulletin of the Academy Sciences, USSR, Math*, 3:3–14, 1941.

[23] Danie G. Krige. *A statistical approach to some mine valuations and allied problems at the Witwatersrand*. University of Witwatersrand, 1951.

[24] Jian Lan and X Rong Li. Tracking of extended object or target group using random matrix—part i: New model and approach. In *2012 15th International Conference on Information Fusion*, pages 2177–2184. IEEE, 2012.

[25] Balmohan V Limaye. *Linear functional analysis for scientists and engineers*. Springer, 2016.

[26] Ronald PS Mahler. *Statistical multisource-multitarget information fusion*. Artech House, Inc., 2007.

[27] Martin Michaelis, Philipp Berthold, Thorsten Luettel, Daniel Meissner, and Hans-Joachim Wuensche. Extended object tracking with an improved measurement-to-contour association.

[28] Hossein Mohammadi, Rodolphe Le Riche, Nicolas Durrande, Eric Touboul, and Xavier Bay. An analytic comparison of regularization methods for gaussian processes. *arXiv preprint arXiv:1602.00853*, 2016.

[29] Katta G Murty. Letter to the editor—an algorithm for ranking all the assignments in order of increasing cost. *Operations research*, 16(3):682–687, 1968.

[30] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press: Cambridge, MA, USA, 2006. Last accessed: 09.08.20.

[31] Kristian Amundsen Ruud. *LIDAR Extended Object Tracking of a Maritime Vessel Using an Ellipsoidal Countor Model*. Norwegian University of Science and Technology, 2018.

[32] Kristian Amundsen Ruud, Edmund Førland Brekke, and Jo Eidsvik. Lidar extended object tracking of a maritime vessel using an ellipsoidal contour model. In *2018 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, pages 1–6. IEEE, 2018.

[33] Ørnulf Jan Rødseth, Svein David Medhaug, and Trond Langemyr. Autonome skip er fremdeles en god idé! `https://www.tu.no/artikler/autonome-skip-er-fremdeles-en-god-ide/491517`, May 2020. Last accessed: 09.08.20.

[34] Michael Schuster and Johannes Reuter. Target tracking in marine environment using automotive radar and laser range sensor. In *2015 20th International Conference on Methods and Models in Automation and Robotics (MMAR)*, pages 965–970. IEEE, 2015.

[35] Michael Schuster, Johannes Reuter, and Gerd Wanielik. Probabilistic data association for tracking extended targets under clutter using random matrices. In *2015 18th International Conference on Information Fusion (Fusion)*, pages 961–968. IEEE, 2015.

[36] Lars-Christian Ness Tokle. *Multi target tracking using random finite sets with a hybrid state space and approximations*. Norwegian University of Science and Technology, 2018.

[37] Niklas Wahlström and Emre Özkan. Extended target tracking using gaussian processes. *IEEE Transactions on Signal Processing*, 63(16):4165–4178, 2015.

[38] Norbert Wiener. *Extrapolation, interpolation, and smoothing of stationary time series*. MIT press Cambridge, MA, 1949.

[39] Jason L Williams. Marginal multi-bernoulli filters: Rfs derivation of mht, jipda, and association-based member. *IEEE Transactions on Aerospace and Electronic Systems*, 51(3):1664–1687, 2015.

[40] Erik F Wilthil, Andreas L Flåten, and Edmund F Brekke. A target tracking system for asv collision avoidance based on the pdaf. In *Sensing and Control for Autonomous Vehicles*, pages 269–288. Springer, 2017.

[41] Yuxuan Xia. Extended-Target-PMBM-Tracker. `https://github.com/yuhsuansia/Extended-Target-PMBM-Tracker`. Online. Last accessed: 09.08.20.

[42] Yuxuan Xia, Karl Granström, Lennart Svensson, Ángel F García-Fernández, and Jason L Williams. Extended target poisson multi-bernoulli mixture trackers based on sets of trajectories. In *2019 22th International Conference on Information Fusion (FUSION)*, pages 1–8. IEEE, 2019.