# Random Finite Set Filters for Superpositional Sensors

Application to Multi–Object Filtering

**A Dissertation**

submitted to the Department of Electrical Engineering and Information Technology of TU Dortmund University in partial fulfillment of the requirements for the degree of

DOKTOR DER INGENIEURWISSENSCHAFTEN
(Doctor of Engineering)

by

Daniel Hauschildt

Dortmund, 2017

**Department of Electrical Engineering and Information Technology**

**Location:**
Dortmund

**Oral Exam:**
06.09.2017

**Supervisors:**
Prof. Dr.–Ing. Uwe Schwiegelshohn
Prof. Dr. Daniel Clark

**Author:**
Daniel Hauschildt

**Title:**
Random Finite Set Filters for Superpositional Sensors

# Abstract

The multi–object filtering problem is a generalization of the well–known single–object filtering problem. In essence, multi–object filtering is concerned with the joint estimation of the unknown and time–varying number of objects and the state of each of these objects. The filtering problem becomes particular challenging when the number of objects cannot be inferred from the collected observations and when no association between an observation and an object is possible.

A rather new and promising approach to multi–object filtering is based on the principles of finite set statistics (FISST). FISST is a methodology, originally proposed by R. Mahler, that allows the formulation of the multi–object filtering problem in a mathematical rigorous way. One of the main building blocks of this methodology are random finite sets (RFSs), which are essentially finite set (FS)–valued random variables (RVs). Hence, a RFS is a RV which is not only random in the values of each element but also random in the number of elements of the FS. Under the premise that the observations are generated by detection–type sensors, many practical and efficient multi–object filters have been proposed. In general, detection–type sensors are assumed to generate observations that either originate from a *single* object or are a false alarms. While this is a reasonable assumption in many multi–object filtering scenarios, this is not always the case.

Central to this thesis is another type of sensors, the superpositional (SPS)–type sensors. Those types of sensors are assumed to generate only one single observation that encapsulates the information about all the objects in the monitored area. More specifically, a single SPS observation is comprised out of the additive contribution of all the observations which would be generated by each object individually. In this thesis multi–object filters for SPS–type sensors are derived in a formal mathematical manner using the methodology of FISST.

The first key contribution is a formulation of a SPS sensor model that, alongside errors like sensor noise, accounts for the fact that an object might not be visible to a sensor due to being outside of the sensor's restricted field of view (FOV) or because it is occluded by obstacles. The second key contribution is the derivation of multi–object Bayes filter for SPS sensors that incorporates the aforementioned SPS sensor model. The third key contribution is the formulation of a filter variant that incorporates a multi–object multi–Bernoulli distribution as underlying multi–object state distribution, thus providing a multi–object multi–Bernoulli (MeMBer) filter variant for SPS–type sensors. As the stated variant turns out not to be conjugate, two approximations to the exact solution are given. The fourth key contribution is the derivation of computationally tractable implementations of the SPS MeMBer filters.

# Declaration

I hereby certify that to the best of my knowledge:

1. This dissertation comprises my original work.
2. Due acknowledgment has otherwise been made to all other material used.
3. This dissertation has not previously been accepted for any degree at TU Dortmund university or another institution.

Daniel Hauschildt

Date

# Eidesstattliche Versicherung

Ich versichere an Eides statt:

1. Diese Dissertation ist selbständig verfasst.
2. Alle in Anspruch genommenen Quellen und Hilfen in der Dissertation wurden vermerkt.
3. Diese Dissertation ist in der vorgelegten oder einer ähnlichen Fassung noch nicht zu einem früheren Zeitpunkt an der TU Dortmund oder einer anderen in- oder ausländischen Hochschule als Dissertation eingereicht worden.

| | |
|---|---|
| Daniel Hauschildt | Datum |

# Acknowledgement

Six years of intense research on a dedicated topic are foremost a lesson of patience, endurance, commitment and dedication. It is not only a trip to the edges of human knowledge but also a trip to and beyond personal limits. In order to successfully pursue this journey of scientific and personal development, an environment that gives support, provides guidance and grants freedom is necessary.

Therefore, I would like to thank my adviser, Prof. Dr.–Ing. Schwiegelshohn, for giving me the time and the freedom to pursue the topic of my interest and providing the environment to endure such a journey. Further, I would like to thank everybody that accompanied me on this journey. It doesn't matter if it was only for a small time or the whole way. More specifically I would like to thank my colleagues at the Robotics Research Institute for all the interesting and enjoyable discussions, all the good times and the patience while listening to my theories. Additionally, I would like to thank my family and friends for supporting me all these years.

# Contents

# List of Acronyms

| | |
|---|---|
| $\Sigma$–MeMBer | Superpositional multi–object multi–Bernoulli |
| AQG | Approximate Quasi Gaussian |
| AQGM | Approximate Quasi Gaussian Mixture |
| AQPB | Approximate Quasi Poisson Binomial |
| CB-MeMBer | Cardinality balanced multi–object multi–Bernoulli |
| CESF | Convoluted elementary symmetric function |
| CPHD | Cardinalized probability hypothesis density |
| DFT | Discrete Fourier Transform |
| EKF | Extended Kalman filter |
| ESF | Elementary symmetric function |
| EV | Expected value |
| FFT | Fast Fourier Transform |
| FISST | Finite set statistics |
| FOV | Field of view |
| FS | Finite set |
| GM | Gaussian mixture |
| GNN | Global Nearest Neighbour |
| GP | Gaussian process |
| IID | Independent and identically distributed |
| IIDC | Independent and identically distributed cluster |
| IQG | Intensity Quasi Gaussian |

| | |
|---|---|
| IQGM | Intensity Quasi Gaussian Mixture |
| IQPB | Intensity Quasi Poisson Binomial |
| JMoG | Joint mixture of Gaussians |
| JPDA | Joint Probabilistic Data Association |
| L-MeMBer | Labeled multi–object multi–Bernoulli |
| MAP | Maximum A Posteriori |
| MC | Monte Carlo |
| MeMBer | Multi–object multi–Bernoulli |
| MGF | Moment generating function |
| MHT | Multi Hypothesis Tracking |
| MoG | Mixture of Gaussians |
| MUD | Multi–User Detection |
| OSPA | Optimal Subpattern Assignment |
| PDA | Probabilistic Data Association |
| PDF | Probability density function |
| PF | Particle filter |
| PGF | Probability generating function |
| PGFL | Probability generating functional |
| PHD | Probability hypothesis density |
| PMF | Probability mass function |
| PPT | Point Process Theory |
| RFS | Random finite set |
| RV | Random variable |
| SCPHD | Single–cluster probability hypothesis density |
| SLAM | Simultaneous Localization and Mapping |

| | |
|---|---|
| SLAT | Simultaneous Localization and Tracking |
| SMC | Sequential Monte Carlo |
| SPS | Superpositional |
| TDOA | Time Difference of Arrival |
| TNC | Thouin–Nannuru–Coates |
| UKF | Unscented Kalman filter |

# List of Symbols

**State Vectors**

| | |
|---|---|
| $x$ | Vector–valued object state |
| $z$ | Vector–valued measurement |
| $\dot{x}$ | Meta object state |
| $x_{k\|k-1}$ | Vector–valued a priori object state at timestep $k$ |
| $x_{k\|k}$ | Vector–valued a posteriori object state at timestep $k$ |
| $x_B$ | Vector–valued new born object state |
| $z_k$ | Vector–valued measurement at timestep $k$ |

**Random Vectors**

| | |
|---|---|
| $\mathbf{x}$ | Random–valued object state |
| $\mathbf{z}$ | Random–valued measurement |
| $\mathbf{w}$ | Random–valued additive noise |

**Finite Sets**

| | |
|---|---|
| $X$ | Finite set–valued object state |
| $X'$ | Finite set–valued a priori object state |
| $X_B$ | Finite set–valued new born object state |
| $Z$ | Finite set–valued measurement |
| $Z_C$ | Finite set–valued clutter measurement |

**Random Finite Sets**

| | |
|---|---|
| $\mathbf{X}$ | RFS–valued object state |
| $\mathbf{Z}$ | RFS–valued measurement state |
| $\mathbf{Z}_C$ | RFS–valued object clutter measurement state |

**Collections and Sequences**

| | |
|---|---|
| $X_{1:k}$ | Time series of finite set–valued object states |
| $Z_{1:k}$ | Time series of finite set–valued measurements |
| $a_0, a_1, \ldots$ | Infinite sequence of values |

**State Spaces**

| | |
|---|---|
| $\mathbb{R}$ | Real numbers |
| $\mathbb{N}$ | Natural numbers |
| $\mathbb{X}$ | Object state space |
| $\mathbb{Z}$ | Measurement space |
| $\mathcal{P}(\mathbb{X})$ | All subsets of the object state space |
| $\mathcal{P}(\mathbb{Z})$ | All subsets of the measurement space |

**Functions and Functionals**

| | |
|---|---|
| $\eta(x)$ | Measurement function |
| $\tau(x)$ | Transition function |
| $f(x)$ | Vector–valued function |
| $f(X)$ | Finite set–valued function |
| $f[h]$ | Vector–valued Functional |
| $F[h]$ | Finite set–valued functional |
| $h(x)$ | Test function in object object state space |
| $h^X$ | Set exponential test function in object space |
| $g(z)$ | Test function in measurement space |
| $g^Z$ | Set exponential test function in measurement space |

**Special Functions**

| | |
|---|---|
| $\delta(x)$ | Dirac delta function |
| $\delta_a(x)$ | Dirac delta function concentrated at $a$ |
| $\sigma_{n,k}(f_1(x), \ldots, f_n(x))$ | Elementary symmetric function (ESF) |

| | |
|---|---|
| $\sigma^{*}_{n,k}(f_1(x),\ldots,f_n(x))$ | Convoluted elementary symmetric function (CESF) |
| $e^x$ or $\exp(x)$ | Exponential function |
| $\theta(i)$ | Mapping function |
| $\Pi_n$ | All possible permutation of length $n$ |

**Operators**

| | |
|---|---|
| $\int f(x)\,\mathrm{d}x$ | Ordinary integral |
| $\int\cdots\int f(x_1,\ldots,x_n)\,\mathrm{d}x_1\cdots\mathrm{d}x_n$ | Ordinary multi–variate integral |
| $\int f(X)\,\delta X$ | Finite set integral |
| $\int\cdots\int f(\{x_1,\ldots,x_n\})\,\mathrm{d}x_1\cdots\mathrm{d}x_n$ | Ordinary multi–variate integral over set–valued function |
| $\frac{\mathrm{d}f(x)}{\mathrm{d}x}$ | Ordinary derivative |
| $\frac{\delta f(X)}{\delta X}$ | Finite set derivative |
| $\frac{\delta F[h]}{\delta X}$ | Functional derivative |
| $n!$ | Factorial |
| $f(1)$ | Evaluation of function $f(x)$ at $x=1$ |
| $F[1]$ | Evaluation of functional $F[h]$ at $h=1$ |
| $\mathbb{E}_f[g(x)]$ | Expectation/Expected value of $g(x)$ with probability density function (PDF) $f(x)$ |

**Binary Operators**

| | |
|---|---|
| $(f_0 * f_1)(x)$ | Convolution of $f_0(x)$ and $f_1(x)$ |
| $f^{*n}(x)$ | Convolution power |
| $f^{(*X)}(x)$ | Convolution set–power |

**Set Operators**

| | |
|---|---|
| $\lvert X\rvert$ | Cardinality of a set $X$ |
| $\mathcal{P}(\cdot)$ | Power set |
| $X_0 \cup X_1$ | Union of finite sets $X_0$ and $X_1$ |
| $X_0 \cap X_1$ | Intersection of finite sets $X_0$ and $X_1$ |

| | |
|---|---|
| $X \subseteq Y$ | $X$ is a proper subset of finite sets $Y$ |

**Special Symbols**

| | |
|---|---|
| $\emptyset$ | Empty set |
| $\theta(i)$ | Association function |
| $\epsilon$ | An arbitrarily small positive quantity |

**Distributions**

| | |
|---|---|
| $\mathcal{N}$ | Normal distribution |
| $\mathcal{B}$ | Bernoulli distribution |
| $\mathcal{MB}$ | Multi–Bernoulli distribution |
| Pois | Poisson distribution |
| $\mathcal{PB}$ | Poisson Binomial distribution |

**Probabilities**

| | |
|---|---|
| $p(n)$ | Cardinality distribution or probability mass function (PMF) |
| $p_S(x)$ | Survival probability |
| $p_D(x)$ | Detection probability |
| $p_V(x)$ | Visibility probability |
| $q$ | Existence probability |
| $q_{\mathrm{B}}$ | Existence probability of new born object |
| $q_{\mathrm{T}}$ | Existence probability of transitioned object |
| $q_d$ | Existence probability of detected object |
| $q_m$ | Existence probability of missed object |
| $q_v$ | Existence probability of visible object |

**Constants and Parameters**

| | |
|---|---|
| $\lambda$ | Rate of a Poisson distribution |
| $\lambda_{\mathrm{B}}$ | Birth rate of Poisson birth distribution |
| $\lambda_C$ | Clutter rate of Poisson clutter distribution |
| $\nu$ | Number of components |

| | |
|---|---|
| $\nu_{k\|k-1}$ | A priori number of components |
| $\nu_{k\|k}$ | A posteriori number of components |
| $\nu_{\mathrm{B}}$ | Number of birth components |
| $K$ | Normalization constant |

**Probability Densities**

| | |
|---|---|
| $s(x)$ | Single–object spatial PDF |
| $s_{k\|k-1}(x)$ | Single–object a priori spatial PDF |
| $s_{k\|k}(x)$ | Single–object a posteriori spatial PDF |
| $s(x)$ | Single–object spatial PDF |
| $s_{\mathrm{B}}(x)$ | Single–object spatial PDF of new born object |
| $s_{\mathrm{T}}(x)$ | Single–object spatial PDF of transitioned object |
| $s_d(x)$ | Single–object spatial PDF of detected object |
| $s_m(x)$ | Single–object spatial PDF of missed object |
| $s_v(x)$ | Single–object spatial PDF of visible object |
| $f_{\mathrm{T}}(x \mid x')$ | Single–object transition PDF |
| $f_{\mathrm{T}}(X \mid X')$ | Multi–object transition PDF |
| $f_{\mathrm{M}}(X \mid X')$ | Multi–object Markov PDF |
| $f_w(z)$ | Single–measurement white noise PDF |
| $f_c(z)$ | Single–measurement clutter PDF |
| $f_c(Z)$ | Multi–measurement clutter PDF |

**Likelihoods**

| | |
|---|---|
| $f(z \mid x)$ | Single–object single–measurement likelihood |
| $f(z \mid X)$ | Multi–object single–measurement likelihood |
| $f(Z \mid X)$ | Multi–object multi–measurement likelihood |
| $\Psi(z \mid x)$ | Single–object single–measurement likelihood of either missing or detecting an object |
| $\Psi^*(z \mid X)$ | Multi–object single–measurement likelihood of either missing or detecting an object. Result of the convolution of $\Psi(z \mid x)$ for all $x \in X$ |

| | |
|---|---|
| $\phi(z)$ | Single–measurement Pseudo–likelihood |
| $L(z-\eta(x))$ | Single–object single–measurement likelihood |

**PGFs and PGFLs**

| | |
|---|---|
| $G(x)$ | Single–object Probability generating function (PGF) of an arbitrary PMF |
| $s[h]$ | Single–object spatial probability generating functional (PGFL) |
| $f[g \mid x]$ | Single–object single–measurement PGFL |
| $f_C[g]$ | Single–measurement clutter PGFL |
| $F[h]$ | Single–object functional |
| $G[h]$ | Multi–object PGFL |
| $G_B[h]$ | Multi–object birth PGFL |
| $G_T[h \mid X]$ | Multi–object transition PGFL |
| $G_M[h \mid X]$ | Multi–object Markov PGFL |
| $G[g \mid X]$ | Multi–Object measurement PGFL |
| $G_D[g \mid X]$ | Multi–Object detection PGFL |
| $G_C[g]$ | Multi–object clutter PGFL |
| $G_m[h]$ | PGFL of the new components induced by missed objects |
| $G_d[h \mid z]$ | PGFL of a detected or visible object induced by a single measurement |
| $G_d[h \mid Z]$ | PGFL of detected or visible objects induced by a multiple measurements |

**Moments**

| | |
|---|---|
| $N$ | Mean cardinality |
| $\sigma^2$ | Variance (univariate) |
| $P$ | Variance (multivariate) |
| $\mu$ | Mean |
| $\mu_i$ | Raw moment of $i$-th order |

| | |
|---|---|
| $\overline{\mu}_i$ | Central moment of $i$-th order |
| $\mu'_i$ | Factorial moment of $i$-th order |
| $M(t)$ | Moment generating function (MGF) |
| $D(x)$ | Probability hypothesis density (PHD) |
| $D_i(x)$ | PHD of $i$-th order |

# List of Figures

# List of Tables

# 1. Introduction to this Thesis

## Contents



## Introduction

On October 7, 2009 Rudolf E. Kalman was awarded the 2008 National Medal of Science from the current US President Barack Obama for his Kalman filter. Thereby acknowledging that almost 50 years after the first presentation of the Kalman filter, it is nowadays undebatable one of the most applied and taught mathematical techniques for removing the noise of time series of data. Most importantly, it allows the estimation of the system state over time from incomplete and erroneous data.

The *data* is often coming from sensors that are used to observe the state of an object. The Kalman filter is then used to estimate the state of the object of interest and infer the value of an unobserved state of a single object. In general this process is done in two steps. In the first step the knowledge of the probable change of the object state over a specific amount of time is used to *predict* the next state of the system, thus its called the prediction step. In the second step this predicted state is corrected using the noisy measurements of the sensors, thus resulting in a presumably more accurate corrected state, therefore it is called the *correction* step. Repeating this process recursively over time then allows the tracking of the state of an object accurately.

However, the Kalman filter is only applicable when the state prediction and correction equations are linear and the noise is Gaussian distributed. Therefore, a more general approach to object state estimation is required.

It is common knowledge that the Bayes filter is a generalized probabilistic approach to estimate an unknown probability density function (PDF) given noisy measurements. The Kalman filter can be seen as a realization of the Bayes filter

when the unknown PDF and object transition PDF and measurement likelihood are all linear Gaussian. For the more general cases where all PDFs are arbitrary and non–linear, different realizations of the Bayes filter are known. Most often used are Sequential Monte Carlo (SMC) realizations which are also called Particle filters.

While these tools are well known to engineers for estimating the state of a single object, another problem arises when there is not only one object state but multiple object states to estimate. Assuming that the sensor is for example a radar sensor monitoring the airspace for aircraft, the sensor will provide multiple measurements depending on the number of aircraft in the monitored area. Under some circumstances, it might be possible that aircraft have some kind of identification mechanism and that is why a measurement could easily be assigned to an aircraft/object. That is why, it would be possible to just have multiple Bayes filters in parallel, one for each object. In practice the problem is often worse as there is generally no object identification. Consequently, no direct association of a measurement and an object is possible and so basically every possible association has to be tried and evaluated. Getting back to the aircraft example there are also other effects that have to be accounted for. First of all, the number of aircraft's in the monitored area may change over time, which may easily be handled if this was the sole effect as the number of measurements would reflect the number of objects. However, this is not true in general. It might happen that some objects are not detected by the sensor and several measurements might not originate from an object. That is why, the number of measurements does not always correspond to the true number of objects.

Multi–object filtering has the aim to solve the aforementioned issues. Hence, multi–object filters have the task to estimate the state and the number of multiple objects simultaneously over time. Recalling the aforementioned, it is acknowledgeable that multi–object state estimation is a considerably hard problem and known to be intractable in general. Various approaches have been extensively studied, most of them rely on some kind of data association between one measurement and one object. The most widely applied approaches are the Global Nearest Neighbour (GNN) (see [Bla86]), Joint Probabilistic Data Association (JPDA) (see [BF88]) and Multi Hypothesis Tracking (MHT) (see [Rei79]). Essentially all these filters rely on the same principles and all have in common that they are basically keeping multiple instances of single–object Bayes filters for all possible objects. In turn, the multi–object filter is basically build out of multiple parallel single–object filters. In order to assign a measurement to a certain object and single–object filter, they predict likely measurements for each single object and then try to associate each predicted measurement with a true measurement by trying to minimize a total cost function. Once an *optimal* match is found the single–object filters are applied accordingly. However, while the assignment can be *optimal* in sense of the cost function, it does not have to be the right choice and thus may lead to subsequent errors in the state estimation.

A different approach, which was introduced by Mahler, is the construction of a true multi–object Bayes filter on the basis of finite set statistics (FISST), modeling

the state of multiple objects as a single finite set (FS) valued state. In contrast to the previously mentioned multi–object filters, these multi–object or random finite set (RFS) Bayes filters do not rely on their single–object counter parts and are *not* just an assemble of multiple single–object Bayes filters. These filters are true multi–object Bayes filters. Furthermore, in case of the RFS filters no data association is necessary and as such these filters do not suffer from making wrong associations between objects and measurements. More importantly, the RFS filter theory and FISST provide a mathematical rigorous foundation for multi–object filtering.

Comparable to the single–object Bayes filter, the multi–object Bayes filter is in general not computationally tractable in its more general form. However, multiple tractable realizations have been published in recent years. In this context mentionable realizations are

- the probability hypothesis density (PHD) filter as described in [Mah03],
- the cardinalized probability hypothesis density (CPHD) filter as described in [Mah07a],
- the multi–object multi–Bernoulli (MeMBer) filter as described in [Mah07b],
- the cardinality balanced multi–object multi–Bernoulli (CB-MeMBer) filter as described in [VVC09],
- and the labeled multi–object multi–Bernoulli (L-MeMBer) filter as described in [VV13].

Recalling the earlier aircraft example, it was assumed that each object will generate at most *one* measurement. While at a first glance this may seem obvious, it is not. In order to comprehend this, it is necessary to take a look where these measurements originate from. Usually, the measurements are not directly generated by a sensor, such as a radar, itself. More likely the *raw* sensor data will be a single measurement containing multiple peaks. Using some kind of detection threshold these peaks will then be extracted from the raw measurement, which in turn results in a set of measurements or detections. In Figure 1.1 the described measurement generation process is depicted. One may notice that this process may introduce errors and discards useful information, thus reducing the state estimation accuracy. Moreover, it can be imagined that under certain circumstances the peaks cannot be extracted correctly. Consequently, these filters will introduce large errors or are not applicable at all.

Thus, a multi–object Bayes filter that uses the raw sensor measurement would not only lead to better results but also would make it possible to use those sensors where the measurement separation is not possible in the first place. Considering these raw measurement signals, there is a certain class of sensors that are ubiquitous, the so called superpositional (SPS)–type sensors. As the name implies these class of sensors measure signals which govern the SPS–principle, and as such the raw measurement is comprised out of the additive contribution of all

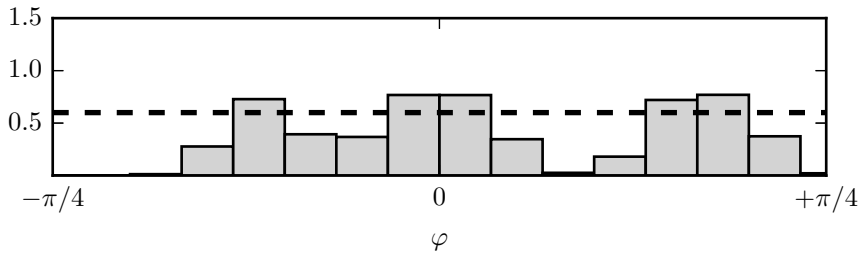


Figure 1.1.: Principle of the measurement generation process. Here, 16 measurements are taken at distinct angles in the range of $\phi = \pm\pi/4$. The raw sensor signal is preprocessed and all peaks over a certain threshold (dotted line) are collected as the set of measurements. Peaks under the threshold will not be detected and as such missed.

individual measurements that would be generated by each object individually. Figure 1.2 illustrates the theoretical generation of such a SPS–type measurement that is generated by three objects. The upper row depicts the individual signals generated by each of these objects and the lower row the resulting raw SPS measurement signal. In this case the resulting raw signal is inseparable.

Considering all this, the question arises if it is possible to derive multi–object Bayes filters that directly operate on the raw SPS measurement. Would they be computationally tractable? Can appropriate computationally tractable approximations be derived in a mathematical rigorous way? Will these filters give good state estimates?

The scope of this thesis is to answer the aforementioned questions. Therefore, appropriate RFS filters for SPS–type sensors will be derived, approximations provided and evaluated. It will be shown that it is indeed possible to derive such filters that are on the one hand computationally tractable and on the other hand give the desired results.

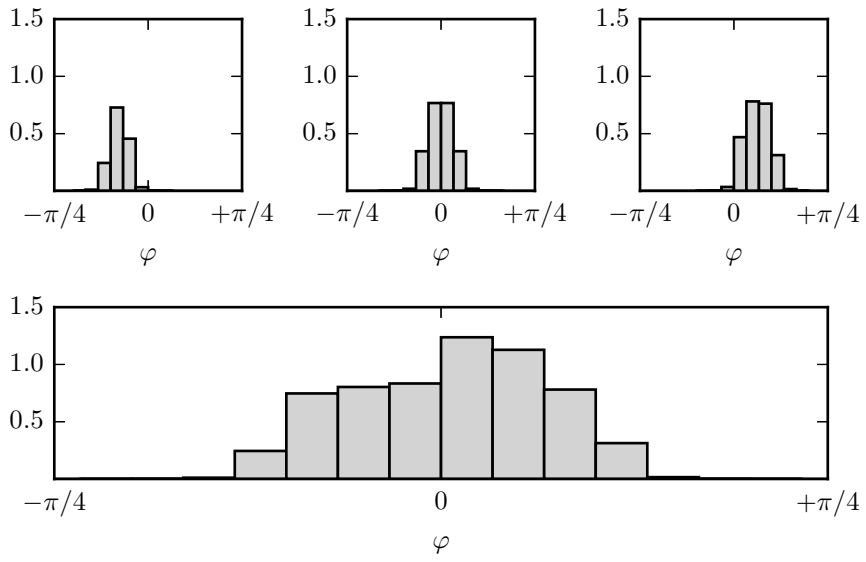


Figure 1.2.: Principle of a SPS–type sensor. Here, 16 measurements are taken at distinct angles in the range of $\phi = \pm\pi/4$. The resulting raw measurement signal is generated by superpositioning the raw signals of three objects. The upper row depicts the individual signals generated by each source and the lower row the superpositioned signal. The resulting raw signal is inseparable.

## 1.1. Motivation

It has been pointed out previously that SPS–type sensors are an important class of sensors. However, not much research has been conducted to use the raw measurement of such sensors for multi–object filtering yet. This is likely due to the fact that it is still considered a hard problem and computationally intractable. Nevertheless, having a multi–object Bayes–filter that can directly work with such sensors would be beneficial and would allow the usage of SPS–type sensors for multi–object state estimation. It may not been noticed at once but SPS–type sensors are ubiquitous and research in this area is important. In the following, several examples of SPS–type sensors will be pointed out in order to emphasize the importance of RFS filters for SPS–type sensors.

### 1.1.1. Thermopile Arrays

A thermopile is a heat sensing sensor that converts heat into voltage. While widely used in infrared thermometers to measure body temperature, Kemper and Hauschildt have repurposed the thermopiles to monitor multiple persons in home environments using their body temperature solely [KH10]. In essence, multiple low–resolution thermopile line arrays are placed at the walls in the monitored area, thus creating a planar two–dimensional view of the area.

Provided that the state of an object is described as $x = (\phi, \Delta, a)^\intercal$ where $\phi$ is the angle of arrival to the sensors center, $\pm\Delta$ is the angular size of the object and $a$ the constant amplitude[1] . Then, the measurement function of a thermopile line array with $m$ pixels is

$$\eta(x) = (\eta_1(x), \ldots, \eta_m(x))^\intercal$$

where the measurement function of the $j$–th pixel is

$$\eta_j(x) = \int_{-\Delta}^{+\Delta} d_j \; a \; cos^b(\alpha + \phi + \theta_j) \; \mathrm{d}\alpha$$

with $d_j$ being a pixel specific damping factor, $\theta_j$ is the displacement angle relative to the angle of arrival $\phi$ and $b$ is a calibration constant depending on the specific sensor.

Assuming that there are $n$ *non–overlapping* objects having states $x_1, \ldots, x_n$ in the sensors field of view (FOV), then the single noisy measurement $\mathbf{z}_j$ provided by a

[1] The amplitude $a$ is proportional to the emitted irradiance of the object. Therefore, it becomes $a \propto \kappa * \left(T_{obj}^4 - T_{sensor}^4\right)$, with $\kappa$ being the Stefan–Boltzmann constant and $T_{obj/sensor}$ are the object and sensor temperatures in Kelvin.

thermopile pixel subject to some additive noise $\mathbf{w}_j$ becomes

$$\begin{aligned}\mathbf{z}_j &= \eta_j(x_1,\ldots,x_n) + \mathbf{w}_j \\ &= \sum_{i=1}^{n} \int_{-\Delta_i}^{+\Delta_i} d_j\, a\, cos^b(\alpha + \phi_i + \theta_j)\, \mathrm{d}\alpha + \mathbf{w}_j.\end{aligned}$$

Concluding, under the aforementioned premises[2] the thermopile array measurement function is SPS. An example of a raw thermopile array measuring the incident irradiance of three non–overlapping objects is visualized in Figure 1.3.

[2] If the objects overlap, then the thermopile model is in fact not a true SPS as the emitted irradiance is in fact blocked and therefore missed by the sensor.

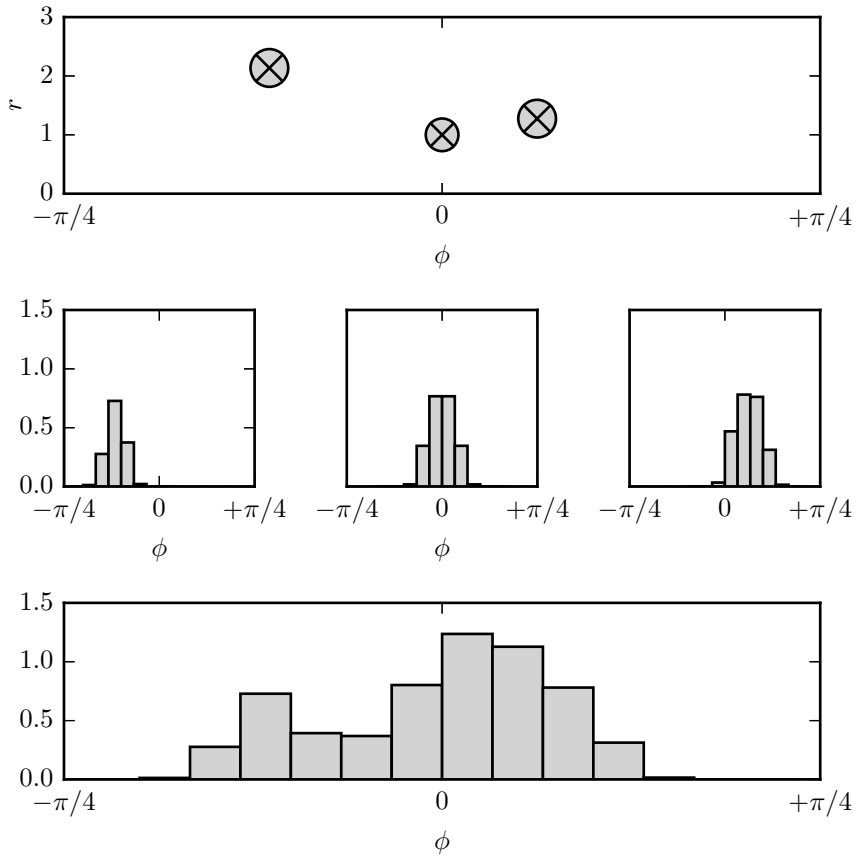


Figure 1.3.: Example of a thermopile array of 16 pixels with parameters $b \approx 284$ and $d = 1$, each measuring the incident irradiance of three objects. The pixel orientations are equally distributed over the range $\pm\pi/4$. The topmost figure depicts the polar position $(r, \phi)$ of the objects, the middle figures show the generated measurements that would be generated by each object individually and the bottommost figure shows the raw SPS–measurement as seen by the thermopile array. It can be seen that the influence of the leftmost object is easily separable but the other two are not.

### 1.1.2. Acoustic Amplitude Sensor

Acoustic Amplitude Sensors are used in tracking scenarios where each object emits an acoustic signal of known amplitude. In [NCM13] Nannuru, Coates, and Mahler placed multiple sensors in the monitored area to track multiple moving objects. Given the emitted constant amplitude $a$, sensor position $p_s$ and an object state $x = (p_o)^\intercal$ where $p_{s/o}$ are the two–dimensional positions of the sensor/object in the Cartesian two–dimensional plane, then the measurement function of $m$ acoustic amplitude sensors becomes

$$\eta(x) = (\eta_1(x), \ldots, \eta_m(x))^\intercal$$

where the measurement function of a single sensor is

$$\eta_j(x) = \frac{a}{\|p_o - p_{s_j}\|^\kappa}$$

with $\kappa$ being a parameter depending on the path loss and $\|\cdot\|$ is the Euclidean norm.

According to [NCM13], the generated measurement $\mathbf{z}_j$ in the presence of multiple objects $x_1, \ldots, x_n$ is

$$\begin{aligned}\mathbf{z}_j &= \eta(x_1, \ldots, x_n) + \mathbf{w} \\ &= \sum_{i=1}^{n} \frac{a_i}{\|p_{o_i} - p_{s_j}\|^\kappa} + \mathbf{w}_j\end{aligned}$$

where $\mathbf{w}_j$ is additive noise. An example of several acoustic amplitude sensors measuring the incident amplitude of three non–overlapping objects is visualized in Figure 1.3.

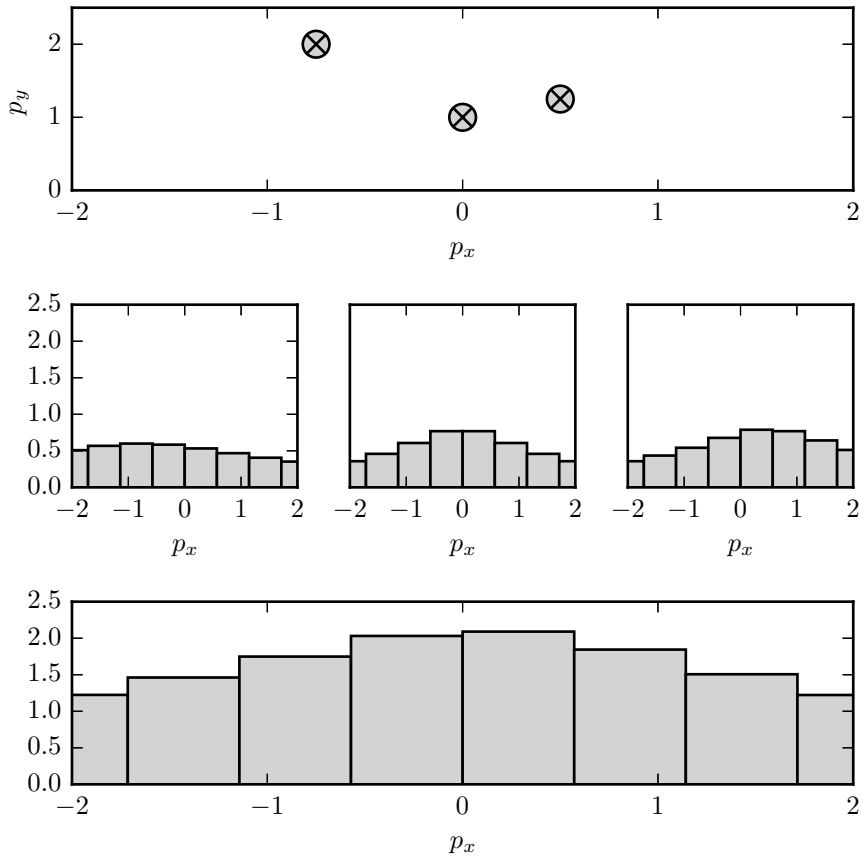


Figure 1.4.: Example of 8 acoustic amplitude sensors measuring the incident amplitude of three objects with $\kappa = 1$. The sensors are placed equally distributed between ±2 on the abscissa. The topmost figure shows the position of the objects in the Cartesian plane $(p_x, p_y)$, the middle figure depicts the resulting measurements for each individual object and the bottommost figure shows the resulting raw SPS–measurement. The signal is clearly inseparable.

### 1.1.3. Other Sensors

In the previous section two examples for SPS–type sensors were demonstrated. In general, a lot more SPS–type sensors exists. In [Mah14, Ch. 19] Mahler shows exemplary that the following applications involve SPS signals under certain assumptions:

1. Surveillance Radar,
2. Positioning using Time Difference of Arrival (TDOA) for sinusoidal signals,
3. Multi–User Detection (MUD) in communication networks
4. and Radio–Frequency tomography.

Moreover, recalling that in physics many forces obey the SPS–principle like for example magnetic, electric or force fields, it can be imagined that there are many possibilities for their usage.

## 1.2. Scope and History

In the previous sections SPS–type sensors have been shown to be ubiquitous. Acknowledging that, it is interesting that most available multi–object state estimation algorithms or tracking filters assume a detection–type measurement model. As mentioned previously, usually the SPS nature of most sensors is neglected and some kind of preprocessing is performed to extract detections out of the raw SPS measurement, thus essentially making the detection–type multi–object filters applicable. However, this introduces errors and thus may perform poorly. Realizing this, multi–object filters that operate directly on the raw SPS measurement could offer a significant improvement over the common, detection–based approaches. In this thesis various novel multi–object Bayes filters for SPS–type sensors based on the RFS approach will be derived and studied. Before that, a brief summary of recent developments in multi–object filtering for detection– and SPS–type sensors is given.

The RFS approach to multi–object Bayes filtering was introduced as a more understandable, more intuitive and thus more applicable way to multi–source multi–object information fusion. While most of the basic results originate from the field of Point Process Theory (PPT), the elegant formulation of FISST has made many improvements in multi–object filtering and information fusion possible in the first place. In the following the key inventions will be considered. Furthermore, for a detailed overview of all advancements in multi–object filtering until 2014 the interested reader is referred to [Mah14].

The first and still most widely known multi–object filter for detection–type sensors is the PHD filter, first presented 2003 in [Mah03]. While it works well, the assumptions made for the distribution of the cardinality of the objects, namely to be Poisson, seemed to be to crude in applications when the object number

gets larger. That is why, the CPHD was introduced 2007 in [Mah07a], basically removing the Poisson restriction and as such having a more stable cardinality estimate.

The PHD class of filters rely on the fact that all potential objects are independent and identically distributed (IID) and as such share the same underlying distribution, thus making it efficient to compute. Nevertheless, another class of filters was introduced as alternative in [Mah07b], denoted MeMBer filters. The MeMBer filters relied on the fact that essentially every object is assigned its own distribution. The first variant was introduced in 2007. Unfortunately, it turned out to have a bias in the cardinality estimate, essentially making it perform much worse than the PHD class of filters. However, in 2009 Vo, Vo, and Cantoni resolved this issue with the publications of the CB-MeMBer filter in [VVC09], thus making it a true alternative to the PHD class of filters. Also, in 2013 a new class of MeMBer filters relying on the notion of labeled RFSs were introduced in [VV13]. These kind of filters are most commonly referred to as L-MeMBer filters.

For each of the mentioned filters different kinds of implementations have been proposed. Most acknowledgeable are the SMC filter implementations like the SMC–PHD [WSG07] in 2007, the SMC–CPHD and the SMC–CB-MeMBer [VVC09] in 2009. Additionally, also analytic implementations like the Gaussian mixture (GM)–PHD [VVC07] in 2007, the GM–CPHD [RCV10] in 2010 or the GM–CB-MeMBer [VVC09] in 2009 have been derived.

In contrast to the SMC implementations which are applicable to any non–linear multi–object filtering problem, the analytic implementations are often only applicable in a linear or mildly non–linear applications when approximation similar to the Extended Kalman filter (EKF) or Unscented Kalman filter (UKF) are applied. This is due to the fact that most analytic implementations of Bayes filters such as the Kalman filter rely on the assumption that all distributions can be modeled by a single Gaussian or a mixtures of Gaussians (MoGs) and that state transition and measurement functions are linear.

All of the aforementioned multi–object Bayes filter realization and implementations have shown to work properly in multi–object tracking scenarios. However, all of these filters assume a detection–type measurement model. While for example a SMC–PHD filter was employed to track multiple moving persons in home environments with thermopile arrays in [KH10], the results are only good as long the raw SPS measurement is easily separable, and as a result only working when the angular distance of the persons or objects is large enough.

Realizing that there is a need for RFS filters for SPS–type sensors, Mahler presented the first CPHD filter realization of such in [Mah09] in 2009. Acknowledging the potential of such a filter, its first analytic implementation was published by Hauschildt in 2011 [Hau11]. Also, the proposed filter was subsequently featured as Hauschildt GM $\Sigma$–CPHD by Mahler in [Mah14]. While in general computable, the Hauschildt $\Sigma$–CPHD is still very computationally demanding and as such only applicable when the number of objects in the monitored area is small. That is why, Mahler and El-Fallah and Nannuru, Coates, and Mahler proposed a computational more tractable approximation to the original $\Sigma$–CPHD and also

introduced the superpositional multi–object multi–Bernoulli ($\Sigma$–MeMBer) filters in [ME12] and [NCM13] called Thouin–Nannuru–Coates (TNC) $\Sigma$–CPHD and –MeMBer filters respectively. Furthermore, Nannuru and Coates also provided a $\Sigma$–MeMBer filter realization in [NC13].

## 1.3. Outline

In this thesis an extensive study of RFS based multi–objects Bayes filters for SPS–type is conducted.

- In Chapter 1, an introduction and brief overview of the history of RFS Bayes filtering and the problem of SPS–type sensors is presented.
- In Chapter 2, an introduction to FISST and multi–object calculus is given. Thereby including the most important statistical descriptors used throughout the thesis.
- In Chapter 3, the most popular RFSs are described, including the Poisson, independent and identically distributed cluster (IIDC) and multi–Bernoulli RFSs.
- In Chapter 4, a more general and extended model for SPS–type sensors and corresponding likelihood is presented. In general, extending the known SPS measurement model previously used throughout literature by considering transmission dropouts and the possibility that some objects may not be visible to or detectable by the sensor.
- In Chapter 5, multi–object Markov transition models are introduced. This includes the modeling of the object motion, appearance and disappearance
- In Chapter 6, multi–object Bayes filters for detection– and SPS–type sensors are presented and their relationship to the single–object variant is pointed out.
- In Chapter 7, MeMBer filter realizations employing the aforementioned SPS measurement model are presented and derived. This includes the exact computationally intractable formulation of the $\Sigma$–MeMBer filter and various efficient approximation making it computationally tractable.
- In Chapter 8, computationally tractable approximations of the previously presented superpositional (SPS) filters are presented. These approximations are applied to the Pseudo–likelihoods introduced in Chapter 7.
- In Chapter 9, SMC implementations of the proposed filters are presented.

Finally, the behavior of the proposed filters is then studied in Chapter 10.

## 1.4. Key Contributions

The key contributions of this thesis are the development of RFS filters for SPS sensors.

- The first key contribution is the derivation of a measurement model for SPS sensors as stated in Section 4.4. In contrast to the conventional model stated in Proposition 3 , the model proposed in Proposition 4 also accounts for the fact that detections might been missed due to objects being occluded or other effects.
- The second key contribution is the deduction of the multi–object Bayes filter corrector equation for the SPS–type sensors as proposed in Proposition 16 in Section 6.2.2.
- The third key contribution is the formulation of the $\Sigma$–MeMBer filter corrector equations as stated in Section 7.2. By substituting the general multi–object distribution in the multi–object Bayes filter corrector with a multi–object multi–Bernoulli distribution, a mathematical rigorous formulation of the corrector equation is derived as stated in Theorems 1 and 2 as well as two approximations. The first approximation stated in Theorem 1 uses approximations similar to the ones used by Vo, Vo, and Cantoni in [VVC07] for the derivation of the CB-MeMBer filter corrector equations. The second approximation stated in Theorem 4 uses the first factorial moment to approximate the exact distribution and derive valid Bernoulli components.
- The fourth key contribution is provided in chapter 8 by deducing computationally tractable approximations. This includes a Gaussian approximation as stated in Theorem 5, a Gaussian Mixture approximation as stated in Theorem 6, and a Poisson Binomial approximation as stated in Theorem 7.

# Part I.

# Calculus of RFS Filters

# 2. Multi–Object Calculus and Finite Set Statistics

## Contents



## Introduction

During the remainder of this thesis the concepts of finite set statistics (FISST) play an important role as they are the foundation of random finite set (RFS)–based multi–object filtering. Therefore, a certain degree of knowledge of set probability theory or more specifically FISST is presumed throughout this thesis. The principles of FISST will later be employed to derive specialized multi–object Bayes filters for tracking the state of multiple objects with superpositional (SPS)–type sensors. The underlying mathematical principles like differentiation and integration, often denoted as *multi–object calculus* in multi–object tracking literature like [Mah07b], are introduced in the following sections.

Note that the following summary of FISST provides only an overview of the most important concepts required for this thesis. A more in–depth introduction to FISST and multi–object calculus and its application is given by Mahler in [Mah04] and [Mah13]. For more details the interested reader should also consider reading [Mah07b].

## 2.1. Random Finite Sets and its Relationship to Multi–Object Filtering

From probability theory most people are familiar with the concept of random integers, random numbers and even random vectors. However, much less common is the concept of a RFS. Basically, RFSs are random variables whose potential instances or values are finite sets (FSs). Equivalent to conventional statistics, which provides useful operations on random integers and numbers, FISST provides operations on RFSs. In essence, RFSs describe collections of random length where each value of an element in the FS is random. Similar to ordinary statistics, where the randomness of a random variable (RV) is described via a probability distribution, the randomness of RFSs is described via set–probability distributions. Some common set–probability distributions will be presented in Chapter 3.

While in conventional single–object filtering random vectors are used to describe the state of an object, RFSs have shown to be a good choice to model the state of a time–varying number of objects as present in multi–object filtering. In this context, the random cardinality of the RFS corresponds to the random number of objects in the monitored area and the random value of an entry of the RFS corresponds to the potential state of a specific object.

It is also common in modern multi–object filtering literature like [Mah07b] to denote the underlying calculus operating on FSs as multi–object calculus and the statistical descriptors provided by FISST as multi–object statistical descriptors like for example multi–object probability density function (PDF), multi–object probability generating functional (PGFL) and so on. Therefore, this terminology will be used throughout this thesis. In the following, the most relevant statistical descriptors like multi–object PDFs, PGFLs, expected values (EVs) and probability hypothesis densities (PHDs) will be presented.

## 2.2. Probability Densities

Multi–object or set probability densities are probability densities of a RFS. As such they describe the relative probability of a RFS to take on the value of a specific FS. In general, multi–object densities are defined as follows. Given a FS $X = \{x_1, \ldots, x_n\}$ of $n \geq 0$ elements, then

$$f(X) = \begin{cases} f(\emptyset) & \text{if } n = 0 \\ f(\{x_1, \ldots, x_n\}) & \text{if } n \geq 1 \end{cases}$$

is a valid multi–object density of the RFS $\mathbf{X}$ if

$$\int f(X)\, \delta X = 1 \text{ and } \forall X \in \mathcal{P}(\mathbb{X}) : \; f(X) \geq 0$$

where $\mathcal{P}(\mathbb{X})$ denotes the power set[1] of the state space $\mathbb{X}$ including the empty set $\emptyset$. Also, it can be noted that $\int f(X)\,\delta X$ is the set–integral as defined in [Mah07b, pp. 360] which expands to an infinite sum over all sets $\{x_1 \dots x_n\}$ of length $n = 0$ to $n = \infty$ and as such is defined as

$$\begin{aligned}\int f(X)\,\delta X &\triangleq \sum_{n=0}^{\infty} \frac{1}{n!} \int\cdots\int f(\{x_1,\dots,x_n\})\,\mathrm{d}x_1 \cdots \mathrm{d}x_n \\ &= f(\emptyset) + \sum_{n=1}^{\infty} \frac{1}{n!} \int\cdots\int f(\{x_1,\dots,x_n\})\,\mathrm{d}x_1 \cdots \mathrm{d}x_n \\ &= \sum_{n=0}^{\infty} \int_{|X|=n} f(X)\,\delta X. \end{aligned} \tag{2.1}$$

Note, that according to [Mah07b, pp. 360] each integral term (2.1) is defined as

$$\int\cdots\int f(\{x_1,\dots,x_n\})\,\mathrm{d}x_1 \cdots \mathrm{d}x_n \triangleq n! \int\cdots\int f(x_1,\dots,x_n)\,\mathrm{d}x_1 \cdots \mathrm{d}x_n$$

where

$$f(x_1,\dots,x_n) = \begin{cases} \frac{1}{n!} f(\{x_1,\dots,x_n\}) & \text{if } x_1,\dots,x_n \text{ are distinct} \\ 0 & \text{else.} \end{cases}$$

Thereby, acknowledging that the probability density $f(x_1,\dots,x_n)$ is the same for each permutation of the variables $x_1,\dots,x_n$. Hence, the probability density can be defined on the FS $\{x_1,\dots,x_n\}$ when taking account for the $\frac{1}{n!}$ possible permutations.

Furthermore, the $n$–th term in equation (2.1) is

$$\int_{|X|=n} f(X)\,\delta X \triangleq \begin{cases} f(\emptyset) & \text{if } n = 0 \\ \frac{1}{n!} \int\cdots\int f(\{x_1,\dots,x_n\})\,\mathrm{d}x_1 \cdots \mathrm{d}x_n & \text{if } n > 0 \end{cases} \tag{2.2}$$

and describes the probability that the RFS $\mathbf{X}$ exactly contains $n$ elements.

## 2.3. Probability Mass Function

While the PDF describes the relative probability of a certain FS, it is often useful to describe the probability of the *cardinality* of FS, the relative probability that the RFS has $n$ members, alone. The straight forward way to deduce this probability

[1] The power set is the state space of all subsets. For example if $\mathbb{X} = \{1, 2, \dots\}$, then $\mathcal{P}(\mathbb{X}) = \{\emptyset, \{1\}, \{2\}, \{1, 2\}, \dots\}$.

mass function (PMF) of a RFS is to recover it from the corresponding PDF $f(X)$ by integrating only over all FSs $X$ with cardinality $|X| = n$ and such resulting in

$$p(n) = \int_{|X|=n} f(X)\, \delta X,$$

where $\int_{|X|=n} f(X)\, \delta X$ is defined in (2.2).

## 2.4. Expected Values

In conventional statistics, the expected value of a continuous RV of a function $g(x)$ given a distribution with PDF $f(x)$ is defined as

$$\mathbb{E}_f[g(x)] \triangleq \int g(x)\, f(x)\, \mathrm{d}x.$$

Similarly, the expected value of a discrete RV with PMF $p(n)$ taking the sequence of values $a = a_0, a_1, \ldots$ is

$$\mathbb{E}_p[a] \triangleq \sum_{n=0}^{\infty} a_n\, p(n).$$

Naturally, The same concept is transferable to RFSs. Given a multi–object PDF $f(X)$ for a RFS $\mathbf{X}$, then the expected value $\mathbb{E}[g(X)]$ of a set–valued function $g(X)$ is

$$\mathbb{E}_f[g(X)] = \int g(X)\, f(X)\, \delta X.$$

However, while one can easily define the expected value, it should be noted that the EV is not defined for every case. Consider for example the simple case where $g(X) \triangleq X$, then $\mathbb{E}_f[X]$ is not defined because the multiplication of a finite–set and a scalar is undefined. Even if it was, the set–integral $\int g(X)\, f(X)\, \delta X$ expands to an infinite sum, each member providing a FS of different length, whose addition is also undefined. But note, as it will be shown later that for some specific choices of $g(X)$, where for example $g : \mathcal{P}(\mathbb{X}) \to \mathbb{X}$ is a reduction[2], a solution might exist.

## 2.5. Probability Generating Functions

Probability generating functions (PGFs) are a useful mathematical tool that ease the derivation of statistical properties in many cases and therefore will largely be

[2] A typical reduction example would be $g(X) \triangleq \sum_{x \in X} g(x)$.

employed in this thesis. Let $p(n)$ be a PMF of a discrete RV, then its PGF $G(y)$ is defined as

$$\begin{aligned} G(y) &\triangleq \mathbb{E}_p\left[y^n\right] \\ &= \sum_{n=0}^{\infty} p(n)\, y^n. \end{aligned}$$

A useful property is that it is possible to derive the PMF directly from the corresponding PGF. So, let $G(y)$ be the PGF of $p(n)$, then the PMF $p(n)$ can be recovered by taking the $n$–th derivative with respect to $y$ and evaluating the result at $y = 0$ such that

$$p(n) = \frac{1}{n!} \frac{\mathrm{d}^n G}{\mathrm{d}y^n}(0).$$

## 2.6. Probability Generating Functionals

While PGFs are defined for discrete RVs having discrete PDFs, the principal idea can be transferred to continuous RVs with continuous PDFs. Given a test function $h(x)$ within the domain $0 \leq h(x) \leq 1$, the probability generating function for a continuous RV $x$ with PDF $f(x)$ is defined as

$$\begin{aligned} f[h] &\triangleq \int h(x)\, f(x) \mathrm{d}x \\ &= \mathbb{E}_f\left[h(x)\right]. \end{aligned}$$

Note that in order to easily distinguish between functions and functionals, functions will be marked by parentheses $(\cdot)$ and functionals with square brackets $[\cdot]$ throughout this thesis. The PGFL of a RFS can be defined similarly. Defining the set power of a function $h(x)$ as

$$h^X \overset{\text{abbr}}{=} \prod_{x \in X} h(x),$$

where the product over a set $X = \{x_1, \ldots, x_n\}$ is defined as

$$\prod_{x \in X} h(x) = \begin{cases} 1 & \text{if } X = \emptyset \\ h(x_1) \cdots h(x_n) & \text{else,} \end{cases}$$

then the PGFL for any RFS given its PDF $f(X)$ becomes

$$G[h] \triangleq \int h^X\, f(X)\, \delta X.$$

It should be noted that there is the connection between PGFL and the corresponding PGF of a RFS. Let $G(y)$ be the PGF of a RFS with underlying PMF $p(n)$,

then the PGF of a multi–object likelihood can be recovered through its PGFL by setting the test function $h(x)$ to a fixed variable $h(x) = y$

$$\begin{aligned} G(y) &= G[y] \\ &= \int y^{|X|} f(X)\, \delta X \\ &= \sum_{n \geq 0} y^n \int_{|X|=n} f(X)\, \delta X \\ &= \sum_{n \geq 0} p(n)\, y^n. \end{aligned}$$

Similar to the previous example, it is also possible to recover the PDF $f(X)$ given the corresponding PGFL.

According to the fundamental theorem of multi–object calculus as stated in [Mah07b, pp. 384] the set–integral and the set–derivative are inverse operations. Expressed in terms of functionals the theorem also states that the underlying function $f(X)$ of a functional $F[h]$ can be recovered by applying the functional derivative $\frac{\delta}{\delta X}$ to the functional $F[h]$, in turn getting a new functional $\left(\frac{\delta F}{\delta X}\right)[h]$ and then setting the test–function $h(x) = 0$. Altogether, the underlying function $f(X)$ can be recovered by $f(X) = \frac{\delta F}{\delta X}[0]$.

Given functional $F[h]$, then the functional derivative [3] is defined as

$$\frac{\delta F[h]}{\delta y} \triangleq \lim_{\epsilon \to 0} \frac{F[h + \epsilon\, \delta_y] - F[h]}{\epsilon},$$

and as such is the derivative in direction of the Dirac delta function $\delta_y(x)$. In turn, the functional derivative with respect to a FS $Y$ is defined as

$$\frac{\delta F[h]}{\delta Y} \triangleq \begin{cases} F[h] & \text{if } X = \emptyset \\ \frac{\delta F[h]}{\delta y_1 \cdots \delta y_n} & \text{if } X = \{y_1, \ldots, y_n\}. \end{cases}$$

As a result, given a PGFL $G[h]$ then the true multi–object PDF can be recovered by

$$f(X) \triangleq \frac{\delta G}{\delta X}[0] \tag{2.3}$$

where $G[0]$ denotes the evaluation of the PGFL $G[h]$ at $h(x) = 0$.

## 2.7. Central, Raw and Factorial Moments

Closely related to the expected values are the moments of an ordinary probability distribution. In ordinary statistics, one distinguishes between several kinds of

[3] A summary of the basic rules for functional derivatives can be found in Appendix A.

moments like the raw, central and factorial moments. Given a discrete probability distribution with PMF $p(n)$, then the $i$–th raw moment is

$$\begin{aligned}\mu_i &\triangleq \mathbb{E}_p\left[n^i\right] \\ &= \sum_{n=0}^{\infty} n^i \; p(n),\end{aligned}$$

the $i$–the central moment is

$$\begin{aligned}\overline{\mu}_i &\triangleq \mathbb{E}_p\left[(n-\mu_1)^i\right] \\ &= \sum_{n=0}^{\infty} (n-\mu_1)^i \; p(n)\end{aligned}$$

and the $i$–th factorial moments is

$$\begin{aligned}\mu_i' &\triangleq \mathbb{E}_p\left[\frac{n!}{(n-i)!}\right] \\ &= \sum_{n=0}^{\infty} \frac{n!}{(n-i)!} \; p(n).\end{aligned}$$

Furthermore, given a continuous probability distribution with PDF $f(x)$, then the $i$–th raw moment is

$$\begin{aligned}\mu_i &\triangleq \mathbb{E}_f\left[x^i\right] \\ &= \int x^i \; f(x) \; \mathrm{d}x,\end{aligned}$$

the $i$–the central moment is

$$\begin{aligned}\overline{\mu}_i &\triangleq \mathbb{E}_f\left[(x-\mu_1)^i\right] \\ &= \int \left(x-\mu_1^i\right) \; f(x) \; \mathrm{d}x\end{aligned}$$

and the $i$–th factorial moments is

$$\begin{aligned}\mu_i' &\triangleq \mathbb{E}_f\left[\frac{x!}{(x-i)!}\right] \\ &= \int \frac{x!}{(x-i)!} \; f(x) \; \mathrm{d}x.\end{aligned}$$

A well known property of the factorial moments is that the $i$-th factorial moment $\mu_i'$ of a discrete probability distribution can be deduced by taking the $i$–th

derivative of the corresponding PGF $G(y)$ with respect to $y$ and substituting $y = 1$. Therefore, the factorial moments are related to a PGF $G(y)$ via

$$\mu_i' = \frac{\mathrm{d}^i G}{\mathrm{d}y^i}(1)$$

Note that the factorial moments $\mu_i'$ are related to the central moments $\overline{\mu}_i$. Therefore, there also exists a relationship between the PGF and the central moments. The relationship of the PGF $G(x)$ to the first two central moments, the mean $N \overset{\text{abbr}}{=} \overline{\mu}_1$ and variance $\sigma^2 \overset{\text{abbr}}{=} \overline{\mu}_2$ , are

$$\begin{aligned} N &= \mu_1' \\ &= \frac{\mathrm{d}G}{\mathrm{d}y}(1) \end{aligned}$$

and

$$\begin{aligned} \sigma^2 &= \mu_2' + \mu_1' - {\mu_1'}^2 \\ &= \frac{\mathrm{d}^2 G(y)}{\mathrm{d}y^2}(1) + \frac{\mathrm{d}G(y)}{\mathrm{d}y}(1) - \left(\frac{\mathrm{d}G(y)}{\mathrm{d}y}(1)\right)^2. \end{aligned}$$

In case of continuous RVs similar relationships exists. Recalling the definition of the PGFL $f[h] = \int h(x)\ f(x)\ \mathrm{d}x$ from Section 2.6, the concept of a moment generating function (MGF) follows if the test–function is set to be $h(x) \triangleq \exp(t^\intercal x)$ with $t, x \in \mathbb{R}^n$ and thus the MGF is defined as

$$\begin{aligned} M(t) &\triangleq f\left[\exp(t^\intercal x)\right] \\ &= \int \exp(t^\intercal x)\ f(x)\ \mathrm{d}x. \end{aligned}$$

Considering this, the factorial moments $\mu_i'$ of a continuous RV can be derived by take the $i$–th derivative from the MGF and setting $t = 0$

$$\mu_i' = \frac{\mathrm{d}^i M}{\mathrm{d}t^i}(0).$$

Unfortunately, there is no direct equivalent to raw, central and factorial moments in FISST. The concept that comes closest to the concept of moments are the moment densities [Mah07b, p. 576].

## 2.8. Factorial Moment Densities and Probability Hypothesis Densities

Note that, as its name implies, the factorial moment densities are not scalar values but densities and as such state dependent. Let $f(X)$ be a multi–object probability

density, $G[h]$ be its PGFL and $X \cup Y$ be the set–union of the FSs $X$ and $Y$, then the $i$–th factorial moment is

$$\begin{aligned} D_i(y_1,\ldots,y_i) &\triangleq \int f(\{y_1,\ldots,y_i\} \cup X)\, \delta X \\ &= \frac{\delta G}{\delta y_1 \ldots \delta y_i}[1]. \end{aligned}$$

For the special cases of $i = 1$ and $i = 2$ the PHD simplifies to

$$\begin{aligned} D_1(y_1) &= \int f(\{y_1\} \cup X)\, \delta X \\ &= \frac{\delta G}{\delta y_1}[1] \end{aligned}$$

and

$$\begin{aligned} D_2(y_1,y_2) &= \int f(\{y_1,y_2\} \cup X)\, \delta X \\ &= \frac{\delta^2 G}{\delta y_1 \delta y_2}[1] \end{aligned}$$

Similar to ordinary statistics there are some moments or moment densities that have special roles. One of these is the PHD, the first order factorial moment $D(y) \stackrel{\text{abbr}}{=} D_1(y)$. Frankly speaking, one can think of the PHD as an unscaled PDF. However, it does not describe the density of probability but the intensity, which is the relative cardinality. As such it does not integrate to 1 but to the expected number of elements in the FS. In the context of multi–object filtering this would correspond to the expected number of objects in the monitored area. Thus, given the PHD $D(y)$ of an arbitrary multi–object PDF $f(X)$, then the expected number of objects – the mean cardinality – $N$ can be determined by

$$N = \int D(y)\, \mathrm{d}y.$$

# 3. Popular Random Finite Sets

## Contents



## Introduction

In Chapter 2 random finite sets (RFSs) were introduced as the building blocks of multi–object calculus. In the following, some important RFSs and some of their relevant statistical properties are summarized. This includes the Poisson RFS (sec. 3.1), the independent and identically distributed cluster (IIDC) RFS (sec. 3.2), the Bernoulli RFS (sec. 3.3) and the multi–Bernoulli RFS (sec. 3.4). A proof of the results summarized in this section can be found in [Mah07b, Ch. 11].

## 3.1. Poisson RFS

A RFS is called Poisson RFS when the cardinality of elements in the set is distributed according to a Poisson distribution with rate $\lambda$ (see [DV03]). Also, the spatial distribution of all elements in the set is assumed to be identically and statistically independent with probability density function (PDF) $s(x)$. As such, the statistical properties of a Poisson RFS are fully described by parameter tuple $(\lambda, s(x))$. Considering this, the statistical properties of a Poisson RFS become as follows. Given the Poisson RFS $X = \{x_1, \ldots, x_n\}$ with parameters $(\lambda, s(x))$, then the multi–object PDF $f(X)$ and probability generating functional (PGFL) $G[h]$ are

$$f(X) \triangleq e^{-\lambda} \prod_{i=1}^{n} \lambda \, s(x_i)$$
$$G[h] = e^{\lambda \, (s[h]-1)},$$

the probability generating function (PGF) $G(x)$, probability mass function (PMF) $p(n)$, the mean $N$ and variance $\sigma^2$ of the cardinality are

$$\begin{aligned} G(y) &= e^{\lambda\,(y-1)} \\ p(n) &= \frac{1}{n!}\lambda^n e^{-\lambda} \\ N &= \lambda \\ \sigma^2 &= \lambda \end{aligned}$$

and the factorial moments/probability hypothesis densities (PHDs) $D_n(x_1,\ldots,x_n)$ are

$$\begin{aligned} D_1(x) &= \lambda\, s(x) \\ D_2(x_1,x_2) &= \lambda^2\, s(x_1)\, s(x_2) \\ D_n(x_1,\ldots,x_n) &= \prod_{i=1}^{n} \lambda\, s(x_i). \end{aligned}$$

## 3.2. Independent and Identically Distributed Cluster RFS

The IIDC RFS can be regarded as generalization of the Poisson RFS described in Section 3.1. If the Poisson assumption for the cardinality distribution is dropped and replaced by an arbitrary discrete cardinality distribution with PMF $p(n)$ the IIDC RFS is then described by the two parameters $(p(n), s(x))$ with $n \triangleq |X|$.

Let $X = \{x_1,\ldots,x_n\}$ be an IIDC RFS with parameters $(p(n), s(x))$, then its multi–object PDF $f(X)$ and PGFL $G[h]$ are

$$\begin{aligned} f(X) &= |X|!\; p\,(|X|) \prod_{i=1}^{|X|} s(x_i) \\ G[h] &= \sum_{n\geq 0} p(n)\; s[h]^n \\ &= G(s[h]), \end{aligned}$$

the PGF $G(y)$, the mean $N$ and variance $\sigma^2$ of the cardinality are

$$\begin{aligned} G(y) &= \sum_{n\geq 0} p(n)\; y^n \\ N &= \sum_{n\geq 0} n\; p(n) \\ \sigma^2 &= \sum_{n\geq 0} n^2\; p(n) - N^2 \end{aligned}$$

and the $n$–th factorial moments $D_n(x_1,\ldots,x_n)$ are

$$\begin{aligned} D_1(x) &= \sum_{k\geq 0} k\; p(k)\; s(x) \\ &= \frac{\mathrm{d}G}{\mathrm{d}y}(1)\; s(x) \\ D_2(x_1,x_2) &= \sum_{k\geq 1} k\;(k-1)\; p(k)\; s(x_1)\; s(x_2) \\ &= \frac{\mathrm{d}^2G}{\mathrm{d}^2y}(1)\; s(x_1)\; s(x_2) \\ D_n(x_1,\ldots,x_n) &= \sum_{k\geq n} \frac{k!}{(k-n)!}\; p(k) \prod_{i=1}^{n} s(x_i) \\ &= \frac{\mathrm{d}^nG}{\mathrm{d}^ny}(1) \prod_{i=1}^{n} s(x_i). \end{aligned}$$

## 3.3. Bernoulli RFS

A Bernoulli RFS is described by two parameters $(q, s(x))$. There is a probability $q$ that an object either does exist or $1-q$ that it does not exist. If an object is existing then the state is distributed according to the probability density $s(x)$. Since a Bernoulli RFS is a singleton and as such can only contain at most one element the finite set $X$ can only have two possible values

$$X = \begin{cases} \emptyset \\ \{x\}. \end{cases}$$

For the Bernoulli RFS the statistical properties are as follows. Given a Bernoulli RFS $\mathbf{X}$ with parameters $(q, s(x))$, then the corresponding probability density $f(X)$, PGFL $G[h]$ and factorial moment/PHD $D_1(x)$ are

$$\begin{aligned} f(X) &\triangleq \begin{cases} 1-q & \text{if } X = \emptyset \\ q\; s(x) & \text{if } X = \{x\} \\ 0 & \text{else} \end{cases} \\ G[h] &= 1 - q + q\; s[h] \\ G(x) &= 1 - q + q\; x \\ D_1(x) &= q\; s(x). \end{aligned}$$

## 3.4. Multi–Bernoulli RFS

The Bernoulli RFS itself is not much useful on its own as it may only contain at most one element. To circumvent this constraint, the multi–Bernoulli RFS $\mathbf{X}$ is a

union of a fixed number $\nu$ of independent Bernoulli RFSs $\mathbf{X}_i$

$$\mathbf{X} = \mathbf{X}_1 \cup \ldots \cup \mathbf{X}_\nu,$$

which can be described by the set of Bernoulli parameters $\{(q_i, s_i(x))_{i=1}^{\nu}\}$. As such it allows the description of up to $\nu$ objects by independent Bernoulli RFSs. Let $\mathbf{X}$ be a multi–Bernoulli RFS with parameters $\{(q_i, s_i(x))_{i=1}^{\nu}\}$ and $X = \{x_1, \ldots, x_n\}$ with $|X| = n$, then according to [Mah07b, pp. 368, 374] its probability density $f(X)$ and PGFL $G[h]$ are

$$\begin{aligned} f(X) &\triangleq \prod_{i=1}^{\nu}(1-q_i) \sum_{1\le i_1\neq\ldots\neq i_n\le\nu} \prod_{j=1}^{n} \frac{q_{i_j}\, s_{i_j}(x_j)}{1-q_{i_j}} \\ &= n! \prod_{i=1}^{\nu}(1-q_i) \sum_{1\le i_1<\ldots<i_n\le\nu} \prod_{j=1}^{n} \frac{q_{i_j}\, s_{i_j}(x_j)}{1-q_{i_j}} \\ G[h] &= \prod_{i=1}^{\nu}(1-q_i+q_i\, s_i[h]) \end{aligned} \tag{3.1}$$

where $\sum_{1\le i_1\neq\ldots\neq i_n\le\nu}$ is the sum taken over all $i_1, \ldots, i_n$ such that $1 \le i_1 \neq \ldots \neq i_n \le \nu$ and $\sum_{1\le i_1<\ldots<i_n\le\nu}$ is the sum taken over all $i_1, \ldots, i_n$ such that $1 \le i_1 < \ldots < i_n \le \nu$.

Further, the PGF $G(x)$, cardinality distribution $p(n)$, the mean $N$ and variance $\sigma^2$ of the cardinality are

$$\begin{aligned} G(y) &= \prod_{i=1}^{\nu}(1-q_i+q_i\, y) \\ p(n) &= \prod_{i=1}^{\nu}(1-q_i) \sum_{1\le i_1<\ldots<i_n\le\nu} \prod_{j=1}^{n} \frac{q_{i_j}}{1-q_{i_j}} \\ &= \prod_{i=1}^{\nu}(1-q_i)\, \sigma_{\nu,n}\left(\frac{q_1}{1-q_1}, \ldots, \frac{q_\nu}{1-q_\nu}\right) \\ N &= \sum_{i=1}^{\nu} q_i \\ \sigma^2 &= \sum_{i=1}^{\nu}(1-q_i)\, q_i \end{aligned}$$

where

$$\sigma_{\nu,n}(x_1, \ldots, x_\nu) = \begin{cases} 1 & \text{if } n = 0 \\ \displaystyle\sum_{1\le i_1<\ldots<i_n\le\nu} \prod_{j=1}^{n} x_{i_j} & \text{if } 1 \le n \le \nu \\ 0 & \text{if } n > \nu \end{cases}$$

denotes the elementary symmetric function (ESF) of degree $n$ in $v$ variables (see [Mac70, pp. 20]), which satisfies the identity

$$\sum_{n=0}^{v} \sigma_{v,n}(x_1,\ldots,x_n) = \prod_{i=1}^{n}(1+x_i).$$

Finally, the $n$–th factorial moments $D_n(x_1,\ldots,x_n)$ are

$$\begin{aligned} D_1(x) &= \sum_{i=1}^{v} q_i \; s_i(x) \\ D_2(x_1,x_2) &= \sum_{1\le i_1\neq i_2\le v} q_{i_1} \; s_{i_1}(x_1)\, q_{i_2} \; s_{i_2}(x_2) \\ &= \sum_{i_1=1}^{v}\sum_{i_2=1}^{v} q_{i_1} \; s_{i_1}(x_1)\, q_{i_2} \; s_{i_2}(x_2) - \sum_{i=1}^{v} q_i^2 \; s_i(x_1)\, s_i(x_2) \\ D_n(x_1,\ldots,x_n) &= \sum_{1\le i_1\neq\ldots\neq i_n\le v} \prod_{j=1}^{n} q_{i_j} \; s_{i_j}(x_j). \end{aligned}$$

# Part II.

# Elements of RFS Filters

# 4. Multi–Object State and Measurement Models

## Contents



## Introduction

In Chapters 2 and 3 the basic concepts of multi–object calculus and random finite sets (RFSs) have been introduced. The focus in this chapter is how these methods can be used to estimate the state of multiple objects under the presence of multiple measurements originating from possibly multiple sensors. In the upcoming sections the following will be shown:

- How to model the state of multiple objects using finite sets (FSs) (sec. 4.1) .
- How to model a collection of multiple measurements as FS (sec. 4.2).

Then, the detection–type measurement model (sec. 4.3) is introduced and the differences to the superpositional (SPS)–type measurement model are pointed out. At last, a model for SPS–type measurements (sec. 4.4) will be derived, starting with the simplest model that has been previously proposed in literature by authors like Mahler in [Mah09] or Nannuru, Coates, and Mahler in [NCM13] and consequently been extended to handle missing detection, missing objects and false alarms. As a result, this leads to a more general measurement model for SPS–type sensors which has not been presented in literature before.

## 4.1. Multi–Object State Space

In single object filtering the objective is to determine the state of a single–object $x$. A common approach to model the state of a single object is to describe its state as a vector $x \in \mathbb{X}$ where all possible values of $x$ are chosen from an arbitrary state space $\mathbb{X}$[1]. Furthermore, in multi object filtering the state of multiple distinct objects has to be described with each having a unique state $x_i$.

Assuming that the number of objects $n$ would be known in advance, then a possible choice of modeling the states would be to introduce a meta state vector

$$\dot{x} \triangleq (x_1^\intercal, \ldots, x_n^\intercal)^\intercal$$

which is composed out of the individual vector–valued object states $x_i$. Unfortunately, considering that the number of objects may vary over time, this approach is not applicable. In contrast to this, the approach favored in finite set statistics (FISST) is to model the state of $n$ individual objects as a single finite set–valued state

$$X = \{x_1, \ldots, x_n\} \in \mathcal{P}(\mathbb{X})$$

where $\mathcal{P}(\mathbb{X})$ denotes the state space of all finite subsets of the single–object state space $\mathbb{X}$ as described in Section 2.2. Note that choosing this particular representation even $X = \emptyset$ is a valid state representing the case where no objects are present.

## 4.2. Multi–Object Measurement Spaces

In the previous section the concept of finite set–valued states was introduced to describe the state of multiple distinct objects. The same concept can be applied for the description of multiple measurements generated by a single or multiple sensors.

Considering that a sensor produces multiple measurements $z_i$, those collections of multiple measurements can conveniently be described by a single finite set–valued measurement

$$Z = \{z_1, \ldots, z_m\} \in \mathcal{P}(\mathbb{Z})$$

consisting of $m$ single–object observation $z_i \in \mathbb{Z}$ where $\mathcal{P}(\mathbb{Z})$ denotes the state space of all finite subsets of the single–object measurement space $\mathbb{Z}$.

## 4.3. Detection–Type Measurement Models

The standard or detection–type measurement model is widely employed to model measurements collected by detection–type sensors. In literature the following assumptions have been postulated as reasonable for these kind of sensors:

[1] In real world scenarios, the values of $x$ are often chosen to be from the $d$ dimensional set of real numbers $\mathbb{X} = \mathbb{R}^d$, however this is not mandatory.

1. A measurement is either generated by a single object or is a false alarm.

2. A single object generates either a single measurement or no measurement at all.

3. All object are points and as such have no physical extend.

4. Measurements are conditionally independent of the objects state.

5. A single sensor observes the scene with an unknown number of unknown states.

The first assumption accounts for the fact that a sensor might have collected some measurements that do not correspond to an actual object and thus must originate from clutter and as such are regarded as false alarms. Note that according to [Mah07b, Sec. 9.3.3.2] the clutter process can be assumed to be approximately Poisson with average clutter rate $\lambda_C$ and single measurement spatial probability density $f_C(z)$.

However, sometimes the clutter process is chosen to be an independent and identically distributed (IID) cluster process with clutter cardinality distribution $p_C(\cdot)$, thus allowing a more fine grained control over the cardinality of the clutter in the expense of computational complexity. While Vo and Mahler used a Poisson Process to derive the efficient equations for the multi–object multi–Bernoulli (MeMBer) in [VVC09] and probability hypothesis density (PHD) filters in [Mah03], Mahler chose to use the independent and identically distributed cluster (IIDC) process to model clutter for the cardinalized probability hypothesis density (CPHD) filter in [Mah07a].

The second assumption represents the fact that some objects in the monitored area might be missed and as such generate no measurement at all. An object will be detected with probability of detection $p_D(x)$ or missed with probability $1 - p_D(x)$.

The third assumption is closely related to the second one. If objects would not be regarded as points as such would have a spatial extend, then there would be a good chance that each object might generate more than one measurement. Also one would have to consider that objects may be hidden by another and as such making the problem of multi–object filtering even more challenging. While this seems to be a huge constraint, this assumption is reasonable in many cases were the distance of the object to the sensor is high or the object extend does not play an important role. In cases were this assumption is not reasonable one has to consider not the *standard* but the so called *extended target* measurement model as described in [Mah07b, pp. 427].

The fourth assumption states that no measurements can be assigned to an object and that is why there is no known association between measurement and an object. Finally, the last assumption just states that there is only a single sensor in use which monitors an unknown number of objects.

### 4.3.1. Detection–Type Measurements

Assuming that a single measurement $z \in \mathbb{Z}$ can be related to an object state $x \in \mathbb{X}$ via the measurement function $\eta : \mathbb{X} \to \mathbb{Z}$, then an object with state $x \in \mathbb{X}$ will generate a measurement

$$z = \eta(x).$$

Considering an error free detection–type sensor that produces no false measurements and missing no detections, then $n$ objects having states $X = \{x_1, \ldots, x_n\}$ would generate a collection of $n$ measurements

$$\begin{aligned} Z &= \{z_1, \ldots, z_n\} \\ &= \{\eta(x_{\pi(1)}), \ldots, \eta(x_{\pi(n)})\}. \end{aligned}$$

Note that unless there is some known association between an object $x_i$ and a measurement $z_j$, then it follows in general that any association is possible, such that the measurement could have been generated by any permutation $\pi : \{1, \ldots, n\} \to \{0, 1, \ldots, n\}$ of the states $x_1, \ldots, x_n$.

However, in the presence of missed detections, the FS of measurements $Z_i$ generated by an object having state $x_i$ would be

$$Z_i = \begin{cases} \emptyset & \text{if the } i\text{-th object is } \textit{not} \text{ detected} \\ \{\eta(x_i)\} & \text{if the } i\text{-th object is detected.} \end{cases}$$

In the detection–type model missed detections represent the possibility that some objects in the monitored area have not generated a measurement. Possible causes are that for most detection–type sensor detections are accepted when they exceed a certain detection–threshold, which is often chosen to be somewhere above the sensor's noise level. It should be noted that in many cases the detection–threshold influences both the amount of missed detections and false alarms/clutter measurements. Evidently, a low detection threshold would result in a low amount of missed detections but high amount of false alarms and vise versa. False alarms or measurements would generate a set of $m_C$ false measurement

$$Z_C = \{z_{C,1}, \ldots, z_{C,m_C}\},$$

and thus the complete set of measurement collected by a detection–type sensor regarding missed detections and false alarms becomes

$$Z = Z_1 \cup \ldots \cup Z_n \cup Z_C.$$

In reality measurements generated by sensors are imperfect and therefore provide inexact results. It is common sense that most of the effects cannot be modeled deterministically and therefore are considered random. Considering that a

single measurement $z$ may be disturbed by random noise modeled by a random variable $\mathbf{w}$, then the measurement is no longer deterministic but also becomes a random vector

$$\mathbf{z} = \eta(x, \mathbf{w}).$$

Assuming that not only the measurement itself are partially random but also the number of measurements is and the set of false measurements is also uncertain described by a RFS $\mathbf{Z}_C$, then the FS of all measurements $Z$ is best described as a RFS

$$\mathbf{Z} = \mathbf{Z}_1 \cup \ldots \cup \mathbf{Z}_n \cup \mathbf{Z}_C$$

where

$$\mathbf{Z}_i = \begin{cases} \emptyset & \text{if the } i\text{-th object is } \textit{not} \text{ detected} \\ \{\eta(x_i, \mathbf{w})\} & \text{if the } i\text{-th object is detected.} \end{cases}$$

Similar to the ordinary single–measurement case one is most likely interested in the likelihood $f(Z \mid X)$ that a certain measurement FS $Z$ is generated by a FS of object state $X$. The most important statistical descriptors for the detection–type measurement model will be summarized in Section 4.3.2.

### 4.3.2. Detection–Type Likelihood and PGFL

Considering the aforementioned assumptions one can derive the likelihood of the standard measurement model with Poisson clutter (sec. 3.1). Here, the main results considering false measurements and missed detections are summarized in Proposition 1.

**Proposition 1.** Let $Z = \{z_1, \ldots, z_m\}$ be the set of $m$ collected measurements, $f(z \mid x)$ be the single–measurement single–object likelihood and $X = \{x_1, \ldots, x_n\}$ be the finite set of $n$ object states. Be $p_D(x)$ an arbitrary probability of detecting a single object given state $x$ and clutter be Poisson with rate $\lambda_C$ and single–measurement probability density function (PDF) $f_C(z)$, then the multi–object likelihood $f(Z \mid X)$ is

$$f(Z \mid X) = \mathrm{e}^{-\lambda_C} f_C(Z) \; f(\emptyset \mid X) \sum_{\theta} \prod_{i:\theta(i)>0} \frac{p_D(x_i) \, f(z_{\theta(i)} \mid x_i)}{(1 - p_D(x_i)) \; \lambda_C f_C(z_{\theta(i)})} \tag{4.1}$$

with

$$f(\emptyset \mid X) \triangleq \prod_{x_i \in X} (1 - p_D(x_i))$$
$$f_C(Z) \triangleq \prod_{z_i \in Z} \lambda_C f_C(z_i).$$

The sum is taken over all possible mappings $\theta$ where

$$\theta : \{1, \ldots, m\} \to \{0, 1, \ldots, n\} \tag{4.2}$$

uniquely associates an element of the measurement set $Z$ with an element of the state set $X$ or no object in the case of a false alarm, marked by the mapping to 0

*Proof.* A formal proof is given in [Mah07b, Ch. 12.3.6] by constructing the equations for the standard measurement model. Starting from the lowest complex variant which does not include any missed detections or clutter, the model is iteratively refined until the equations resemble the ones given in Proposition 1.

Having a closer look at the multi–object likelihood in (4.1) reveals that even under the strong assumptions the complexity is still very high since all possible associations $\theta$ between any collected measurement and any possible state have to be considered, resulting in exponential complexity with upper bound $2^{n+m}$ possible associations[2].

Given the likelihood of the standard model as presented in Proposition 1, then it is also possible to derive its probability generating functional (PGFL).

**Proposition 2.** Let the likelihood of the detection–type measurement model be as stated in Proposition 1, then the corresponding PGFL $G[g \mid X]$ can be factorized in PGFLs for detection and clutter. Hence, it becomes

$$G[g \mid X] = G_C[g]\, G_D[g \mid X] \tag{4.3}$$

where

$$\begin{aligned} G_D[g \mid X] &= \prod_{x \in X} (1 - p_D(x) + p_D(x) f[g \mid x]) \\ f[g \mid x] &= \int g(z)\, f(z \mid x)\, \mathrm{d}z \\ G_C[g] &= \mathrm{e}^{\lambda_C\, (f_C[g] - 1)} \\ f_C[g] &= \int g(z)\, f_C(z)\, \mathrm{d}z \end{aligned}$$

are the PGFLs of the detections $G_D[g \mid X]$, Poisson clutter $G_C[g]$ and single–object likelihoods $f[g \mid X]$ with corresponding single–object single–measurement PDFs $f_C(z)$ and $f(z \mid x)$.

*Proof.* A formal proof of Proposition 2 is performed in [Mah07b, App. G.19] by deriving the PGFL of (4.1) as proposed in Proposition 1.

[2] $\sum_{k=0}^{\min(n,m)} \binom{m}{k} \binom{n}{k} \leq \left( \sum_{k=0}^{\min(n,m)} \binom{m}{k} \right) \left( \sum_{k=0}^{\min(n,m)} \binom{n}{k} \right) \leq \left( \sum_{k=0}^{m} \binom{m}{k} \right) \left( \sum_{k=0}^{n} \binom{n}{k} \right) = 2^{m+n}$

Even more general standard measurement models with any desired state independent clutter model can easily be derived by dropping the Poisson false alarm assumption and replacing the Poisson PGFL $G_C[g] = e^{\lambda_C (f_C[g]-1)}$ in equation (4.3) with the PGFL of an arbitrary clutter model

$$G_C[g] = \int g^Z \, f_C(Z) \, \delta Z,$$

where $f_C(Z)$ can be any multi–object or set–PDF. Also, in [Mah07b] it is proven that it is possible to replace the state–independent false alarm model with a state–dependent by replacing PGFL $G_C[g]$ with a state–dependent variant $G_C[g \mid X]$. In both cases, the corresponding likelihoods can easily be derived following the principles of FISST as described in Chapter 2.

## 4.4. Superposition–Type Measurement Models

The SPS measurement model differs from the detection–type model in one small but important aspect. A SPS–type sensor generates only *one* measurement for a collection of objects in its monitored area. That is why, the standard model can no longer be applied as it breaks one of the major assumptions made in order to derive the standard model. In the following sections a measurement model for SPS–type sensors will be derived. In literature like [Mah09] or [NC13] the proposed SPS measurement model is rather simplistic and does not account for object visibility, missed detections, false alarms or other possible effects similar to the detection–type model.

Here a more complete model, which also accounts for missed detections and clutter, will be developed that underlies the following assumptions:

1. A single sensor observes the scene with an unknown number of $n$ objects each having a distinct unknown state $x_i$.

2. A single measurement $z$ is a composed out of the sum – the superposition – of the individual sub–signals $z_i = \eta(x_i)$ generated by an objects state $x_i$ as $z = \sum_{i=1}^{n} \eta(x_i)$.

3. A single object is either *visible* to or *detectable* by the sensor and as such influences the measurement additively or not at all with a probability $p_V(x_i)$ of visibility.

4. All object states are stochastically independent of each other.

5. The sensor noise is additive and stochastically independent of the objects' states.

6. The objects are points and as such have *no* physical extent.

Most of the aforementioned assumptions are equal to the ones of the standard model. However, it is also presumed that the sensor noise is additive and independent of the object state while in the standard model the noise can be arbitrary in theory. Moreover, in the detection–type model objects visibility is handled by missed detections. However, this is not the case for the SPS–type model. Here, the objects visibility accounts for the fact that the object is or is not in the field of view (FOV) of the SPS sensor or hidden for various other reasons. If the object is not visible to the sensor due to occlusion, then the measurement will not contain any additive contribution that originates from the occluded object. Since the visibility is only depending on the state of the object, the probability of being visible can be modelled similar to the probability of detection as a state dependent probability of visibility $p_V(x)$.

### 4.4.1. Superposition–Type Measurements

In contrast to detection–type sensors, SPS–type sensors will generate a single measurement $z$ which is dependent on the states $X = \{x_1, \ldots, x_n\}$ of all objects in the monitored area. In turn, the finite set of measurement $Z$ consists only of one measurement $Z = \{z\}$. The multi–object measurement function $\eta : \mathcal{P}(\mathbb{X}) \to \mathbb{Z}$ with

$$\eta(X) \triangleq \sum_{i=1}^{n} \eta(x_i) \tag{4.4}$$

relates the object states $X = \{x_1, \ldots, x_n\}$ to the measurement

$$z = \sum_{i=1}^{n} \eta(x_i).$$

Noticing that some object states are not *visible* to the sensor, it might happen that some objects do not influence the measurement $z$ at all. If $z_i = \eta(x_i)$ is a single measurement that would be generated by a single object state $x_i$, then, if object invisibility is considered, the measurement function becomes

$$\hat{\eta}(x_i) = \begin{cases} 0 & \text{if the } i-th \text{ object it } \textit{not} \text{ visible} \\ \eta(x_i) & \text{if the } i-th \text{ object is visible.} \end{cases}$$

Acknowledging that measurements are imperfect and disturbed by additive random noise $\mathbf{w}$ the single SPS measurement $z$ becomes a random variable (RV)

$$\mathbf{z} = \sum_{i=1}^{n} \hat{\eta}(x_i) + \mathbf{w}.$$

### 4.4.2. Superposition–Type Likelihood and PGFL

In the previous section the fundamental differences between the detection–type sensors measurement and a SPS–type sensor's measurement have been stated. In

the following, the likelihoods and the corresponding PGFLs for different cases are presented with increasing complexity.

#### 4.4.2.1. No Missed Objects/Detections

Considering the aforementioned assumptions and neglecting the presence of missed objects, missing detections and other effects, the likelihood for the SPS measurement is presented in the following Proposition 3.

**Proposition 3.** Let $X = \{x_1, \ldots, x_n\}$ be a finite set of object states, $z$ be the SPS measurement and $\eta(X) \triangleq \sum_{i=1}^{n} \eta(x_i)$ with $\eta(x)$ being the single–object measurement function, the SPS likelihood becomes

$$f(Z \mid X) = \begin{cases} f(z \mid X) & \text{if } Z = \{z\} \\ 0 & \text{else} \end{cases} \tag{4.5}$$

with

$$\begin{aligned} f(z \mid X) &= f_w(z - \eta(X)) \\ &= \left(f_w * \delta_{\eta(X)}\right)(z) \\ &= \left(f_w * \delta_{\eta(x_1)} * \ldots * \delta_{\eta(x_n)}\right)(z), \end{aligned} \tag{4.6}$$

where $f_w(z)$ is the PDF of the additive noise vector $\mathbf{w}$, $*$ is the convolution operator and $\delta_{\eta(x)}(z)$ is the Dirac delta function concentrated at $\eta(x)$. Additionally, the PGFL is determined by the ordinary integral

$$\begin{aligned} G[g \mid X] &= \int g^Z f(Z \mid X)\, \delta Z \\ &= \int g(z)\, f_w(z - \eta(X))\, \mathrm{d}z \\ &= \int g(z)\, f_w\left(z - \sum_{x_i \in X} \eta(x_i)\right). \end{aligned}$$

*Proof.* By definition the object states $X = \{x_1, \ldots, x_n\}$ and the noise $\mathbf{w}$ are stochastically independent. It is common knowledge that the PDF of the sum of independent random variables is the convolution of the individual PDFs. So, assuming that $f_w(z)$ is the PDF of the random variable $\mathbf{w}$ and

$$f_i(z) = \int \delta_{\eta(x)}(z)\, f_i(x)\, \mathrm{d}x$$

x is the PDF of the transformation of the RV $\mathbf{x}$ from the state to measurement space, then the PDF $f(z \mid X)$ of the RV $\mathbf{z}$ becomes

$$f(z \mid X) = (f_w * f_1 * \ldots * f_n)(z), \tag{4.7}$$

where $\delta_{\eta(x)}(z)$ is the Dirac delta function concentrated at $\eta(x)$. Considering that the Dirac delta function is defined as

$$\delta_{x_1}(x) \equiv \delta(x - x_1)$$
$$\delta(x) = \begin{cases} \infty & \text{if } x = 0 \\ 0 & \text{else} \end{cases}$$

and the convolution of two arbitrary functions $f_1(x)$ and $f_2(x)$ is defined as

$$(f_1 * f_2)(x) \triangleq \int f_1(x - y)\, f_2(y)\, \mathrm{d}y,$$

then the convolution of two Dirac delta functions results in the identity

$$(\delta_{x_1} * \delta_{x_2})(x) = \delta_{x_1 + x_2}(x).$$

Since each object state $x_i$ is deterministic and not random at all, its PDF is

$$f(x_i) \triangleq \delta_{x_i}(x). \tag{4.8}$$

Inserting (4.8) into (4.7) it follows as claimed

$$\begin{aligned} f(z \mid X) &= (f_w * \delta_{\eta(x_1)} * \ldots * \delta_{\eta(x_n)})(z) \\ &= (f_w * \delta_{\eta(X)})(z) \\ &= f_w(z - \eta(X)) \end{aligned} \tag{4.9}$$

where

$$\begin{aligned} \delta_{\eta(X)}(z) &= \left(\delta_{\eta(x_1)} * \ldots * \delta_{\eta(x_n)}\right)(z) \\ &= \delta_{\sum_{i=1}^{n} \eta(x_i)}(z). \end{aligned}$$

Considering these results, the PGFL can be determined by the set–integral

$$G[g \mid X] = \int g^Z f(Z \mid X)\, \delta Z.$$

By replacing the measurement likelihood by its definition in (4.5), the set–integral simplifies to an ordinary integral and becomes

$$G[g \mid X] = \int g(z)\, f(z \mid X)\, \mathrm{d}z.$$

Finally, after applying the result from (4.9) and the definition of $\eta(X)$ from (4.4), it follows that

$$\begin{aligned} G[g \mid X] &= \int g(z)\, f_w(z - \eta(X))\, \mathrm{d}z \\ &= \int g(z)\, f_w\left(z - \sum_{x_i \in X} \eta\left(x_i\right)\right) \mathrm{d}z. \end{aligned}$$

□

#### 4.4.2.2. Missed Objects/Detections

As described in Section 4.4.1, objects in the monitored area may be missed by the sensor due to being outside of the sensor's restricted FOV or because they are occluded by obstacles. Extending the simplistic model in Section 4.4.2.1 to account for those effects leads to the extended SPS model described in Proposition 4.

**Proposition 4.** Let $Z = \{z\}$ be a SPS measurement, $X = \{x_1, \ldots, x_n\}$ be the finite set of object states, $p_V(x)$ be the single–object probability of visibility and $f_w(z)$ the noise PDF, then the likelihood $f(Z \mid X)$ of a SPS measurement is

$$f(Z \mid X) = \begin{cases} f(z \mid X) & \text{if } Z = \{z\} \\ 0 & \text{else} \end{cases}$$

$$\begin{aligned} f(z \mid X) &= \sum_{X' \subseteq X} (1 - p_V)^{X \setminus X'} \, {p_V}^{X'} \, f_w\left(z - \sum_{x' \in X'} \eta(x')\right) \\ &= (1 - p_V)^X \sum_{X' \subseteq X} \frac{{p_V}^{X'}}{(1 - p_V)^{X'}} \, f_w\left(z - \sum_{x' \in X'} \eta(x')\right) \qquad (4.10) \\ &= (f_w * \Psi_1 <* \ldots * \Psi_n)(z) \qquad (4.11) \end{aligned}$$

where

$${p_V}^X \overset{\text{abbr}}{=} \prod_{x \in X} p_V(x)$$
$$(1 - p_V)^X \overset{\text{abbr}}{=} \prod_{x \in X} (1 - p_V(x))$$

is the set–power and

$$\Psi_i(z \mid x_i) = (1 - p_V(x_i))\, \delta_0(z) + p_V(x_i) \delta_{\eta(x_i)}(z) \qquad (4.12)$$

is the likelihood that a single object either generates an additive contribution of 0 with probability $1 - p_V(x)$ or an additive contribution of $\eta(x)$ with probability $p_V(x)$. Considering the aforementioned likelihood, its corresponding PGFL is

$$\begin{aligned} G[g \mid X] &= \int g(z)\, f(z \mid X)\, \mathrm{d}z \\ &= (1 - p_V)^X \sum_{X' \subseteq X} \frac{{p_V}^{X'}}{(1 - p_V)^{X'}} \, G[g \mid X'] \end{aligned}$$

where $G[g \mid X']$ is the PGFL of the SPS measurement model as stated in Proposition 3.

Most of the aforementioned equations are straightforward. However, the equivalence of (4.10) and (4.11) will be explicitly provided in the following.

*Proof.* First it has to be noted that the sum over all subsets $\sum_{X' \subseteq X}$ can be rewritten to

$$\begin{aligned}\sum_{X' \subseteq X} f(X') &= \sum_{k=0}^{n} \sum_{X' \subseteq X, |X'|=k} f(X') \\ &= \sum_{k=0}^{n} \sum_{1 \leq i_1 < \ldots < i_k \leq n} f(\{x_{i_1}, \ldots, x_{i_k}\}).\end{aligned}$$

Considering this and also replacing the set–power with its definition, then (4.10) becomes

$$f(z \mid X) = (1 - p_V)^X \sum_{k=0}^{n} \sum_{1 \leq i_1 < \ldots < i_k \leq n} \left( \prod_{j=1}^{k} \frac{p_V(x_{i_j})}{1 - p_V(x_{i_j})} \right) f_w \left( z - \sum_{j=1}^{k} \eta(x_{i_j}) \right).$$

Furthermore, it is known from (4.6) that

$$f_w \left( z - \sum_{i=1}^{k} \eta(x_i) \right) \equiv \left( f_w * \delta_{\eta(x_1)} * \ldots * \delta_{\eta(x_k)} \right)(z)$$

and such it follows

$$\begin{aligned} f(z \mid X) &= (1 - p_V)^X \sum_{k=0}^{n} \sum_{1 \leq i_1 < \ldots < i_k \leq n} \left( \prod_{j=1}^{k} \frac{p_V(x_{i_j})}{1 - p_V(x_{i_j})} \right) \left( f_w * \delta_{\eta(x_{i_1})} * \ldots * \delta_{\eta(x_{i_k})} \right)(z) \\ &= (1 - p_V)^X \sum_{k=0}^{n} \sum_{1 \leq i_1 < \ldots < i_k \leq n} \left( f_w * \frac{p_V(x_{i_1}) \delta_{\eta(x_{i_1})}}{1 - p_V(x_{i_1})} * \ldots * \frac{p_V(x_{i_k}) \delta_{\eta(x_{i_k})}}{1 - p_V(x_{i_k})} \right)(z) \\ &= (1 - p_V)^X \left( f_w * \sum_{k=0}^{n} \sum_{1 \leq i_1 < \ldots < i_k \leq n} \frac{p_V(x_{i_1}) \delta_{\eta(x_{i_1})}}{1 - p_V(x_{i_1})} * \ldots * \frac{p_V(x_{i_k}) \delta_{\eta(x_{i_k})}}{1 - p_V(x_{i_k})} \right)(z). \end{aligned} \tag{4.13}$$

Before continuing, it can be noted that the elementary symmetric function (ESF) of degree $k$ in $n$ variables or functions (see [Mac70, pp. 20]) is defined as

$$\sigma_{n,k}(f_1(x), \ldots, f_n(x)) = \begin{cases} 1 & \text{if } k = 0 \\ \sum_{1 \leq i_1 < \ldots < i_k \leq n} \prod_{j=1}^{k} f_{i_j}(x) & \text{if } 1 \leq k \leq n \\ 0 & \text{if } k > n \end{cases}.$$

Hence, the ESF is the sum over all $k$-subsets of the set of the functions $\{f_1, \ldots, f_n\}$, which in turn gives us the sum over all combinations of length $k$. Furthermore, when summing over all possible $0 \leq k \leq n$ this satisfies the identity

$$\prod_{i=1}^{n} (1 + f_i(x)) = \sum_{k=0}^{n} \sigma_{n,k}(f_1(x), \ldots, f_n(x)).$$

Equivalently, the ESF can be defined for the convolution instead of the product. Therefore, the convoluted elementary symmetric function (CESF) can be defined as

$$\sigma^*_{n,k}(f_1(x),\ldots,f_n(x)) \triangleq \begin{cases} \delta_0(x) & \text{if } k=0 \\ \sum\limits_{1\le i_1<\ldots<i_k\le n} \left(f_{i_1} * \ldots * f_{i_k}\right)(x) & \text{if } 1\le k\le n \\ 0 & \text{if } k>n \end{cases}, \tag{4.14}$$

satisfying the identity

$$((\delta_0 + f_1) * \ldots * (\delta_0 + f_n))(x) = \sum_{k=0}^{n} \sigma^*_{n,k}(f_1(x),\ldots,f_n(x)) \tag{4.15}$$

where $(\delta_0 * f)(x) \triangleq f(x)$. Then, after applying this to (4.13) is follows

$$f(z \mid X) = (1-p_V)^X \left(f_w * \sum_{k=0}^{n} \sigma^*_{n,k}\left(\frac{p_V(x_1)\delta_{\eta(x_1)}}{1-p_V(x_1)},\ldots,\frac{p_V(x_n)\delta_{\eta(x_n)}}{1-p_V(x_n)}\right)\right)(z).$$

Consequently, using the identity from (4.15) this becomes

$$\begin{aligned} f(z \mid X) &= (1-p_V)^X \left(f_w * \left(\delta_0 + \frac{p_V(x_1)\delta_{\eta(x_1)}}{1-p_V(x_1)}\right) * \ldots * \left(\delta_0 + \frac{p_V(x_n)\delta_{\eta(x_n)}}{1-p_V(x_n)}\right)\right)(z) \\ &= \left(f_w * \left((1-p_V(x_1))\delta_0 + p_V(x_1)\delta_{\eta(x_1)}\right) * \ldots * \left((1-p_V(x_n))\delta_0 + p_V(x_n)\delta_{\eta(x_n)}\right)\right)(z). \end{aligned}$$

Finally, with

$$\Psi_i(z \mid x_i) \triangleq (1-p_V(x_i))\,\delta_0(z) + p_V(x_i)\delta_{\eta(x_i)}(z)$$

this results in

$$f(z \mid X) = (f_w * \Psi_1 * \ldots * \Psi_n)(z).$$

□

#### 4.4.2.3. False Alarms/Measurements

In the standard model described in Section 4.3 which applies to detection–type sensors, clutter regards to false measurements generated by errors in the detection process. Normally those errors are generated by some artifacts in the signal or some equivalent disturbances. Consequently, a clutter measurement introduces additional *clutter* or *ghost* objects and as such, if not handled appropriately, also introduces a bias in the estimated number of objects. The important fact is that a clutter model can easily be defined solely on the measurement and is in turn independent of any object's state. In the SPS model a clutter model that solely depends on the measurement can only determine if the measurement of the sensor can be regarded as valid or not.

# 5. Multi–Object Markov Models

## Contents



## Introduction

In the single–object scenario the transition of an object having state $x'$ to a new state $x$ is most often modelled by its physical motion. However, in the multi–object case not only the motion of the individual objects has to be modelled but also additional effects like the appearance of new objects in the monitored area or the disappearance of those. In literature one standard model is widely employed that obeys the following assumptions:

1. Objects transition independently of each other.

2. Objects may disappear from the monitored area.

3. New objects may enter the monitored area.

The first assumption regards to the fact that each object moves or transitions independently. It is assumed that there is no interaction between any of these objects like collisions or group transition and so on. The second assumption states that objects in the monitored area may disappear which is often referred to as object death and survival. Object death may happen due to several reasons such as for example that objects may leave the monitored area or objects simply vanish. The third assumption accounts for the possible appearance of new objects in the scene. Object appearance is often denoted by object birth.

## 5.1. Multi–Object State Transition Model

Assuming that there are $n'$ objects in the scene described by a finite set (FS)–valued state $X' = \{x'_1, \ldots, x'_{n'}\}$ and considering that each object transitions independently, then each individual object's transition can be described by the single–object transition function $\tau : \mathbb{X} \to \mathbb{X}$. Then, the new state $x'$ becomes

$$x_i = \tau(x'_i).$$

Consequently, given the FS $X'$, then the FS $X$ after the transition would be

$$\begin{aligned} X &= \{x_1, \ldots, x_n\}. \\ &= \{\tau(x'_1), \ldots, \tau(x'_n)\}. \end{aligned}$$

However, recalling that objects may disappear due to leaving the monitored area or other reasons, the resulting finite set will be

$$X = X_1 \cup \ldots \cup X_n$$

where each finite set $X_i$ is

$$X_i = \begin{cases} \emptyset & \text{if the i–th object has disappeared or died} \\ \{\tau(x'_i)\} & \text{if the i–th object still exists / survives.} \end{cases}$$

Additionally, new objects may enter the scene. Therefore, a variable number of new states may be generated independently of the object states. Thus, if the FS of $n_{\mathrm{B}}$ new born states is

$$X_{\mathrm{B}} = \{x_{\mathrm{B}_1}, \ldots x_{\mathrm{B}_{n_{\mathrm{B}}}}\},$$

then the resulting FS of all states after object transition, disappearance and birth becomes the union of all those individual states and as such follows to be

$$X = X_1 \cup \ldots \cup X_n \cup X_{\mathrm{B}}.$$

Assuming that the transition of an object is not exactly known or cannot be modeled accurately, then there is some uncertainty in the resulting new object state $x_i$ given the previous state $x'_i$. In ordinary probability theory it is common to model this uncertain behavior with transition densities $f(x_i \mid x'_i)$, the probability density that the new state $x_i$ follows when the previous state is $x'_i$. The same principal is true for multiple objects. However, the multi–object transition density defines the probability density that a FS $X = \{x_1, \ldots, x_n\}$ originated from FS $X' = \{x'_1, \ldots, x'_{n'}\}$. Note that the number $n$ of elements in $X$ must not necessarily be equal to the number $n'$ of elements in $X'$ since objects may vanish or be created/born.

## 5.2. Multi–Object State Markov Densities

Recalling the single–object state transition case, the Markov or transition density $f_{\mathrm{M}}(x \mid x') = f_{\mathrm{T}}(x \mid x')$ defines the probability that an object with current state $x'$ has state $x$ afterwards. In essence, the same is true for the multi–object state case, the Markov density $f_{\mathrm{M}}(X \mid X')$ defines the probability that the multiple objects having the finite set–valued state $X' = \{x'_1, \ldots, x'_{n'}\}$ have the finite set–valued state $X = \{x_1, \ldots, x_n\}$ afterwards.

In literature, a common transition model has been establish which will be referred to as *standard* transition model. This standard transition model underlies the following assumptions:

1. Objects move or transition independently and as such the object transition can be modeled by the single–object transition probability density function (PDF) $f_{\mathrm{T}}(x \mid x')$.

2. An object survives with a single–object probability of survival $p_S(x')$ or dies with probability $1 - p_S(x')$.

Under the aforementioned assumption the Markov transition density and its corresponding probability generating functional (PGFL) becomes as proposed in Proposition 5.

**Proposition 5.** Let $f_{\mathrm{T}}(x \mid x')$ be the single–object transition density, $p_S(x')$ be the probability of survival. Be $X = \{x_1, \ldots, x_n\}$ and $X' = \{x'_1, \ldots, x'_{n'}\}$ where $n \leq n'$. Then, the conditional PDF $f_{\mathrm{T}}(X \mid X')$ and its PGFL $G_{\mathrm{T}}[h \mid X']$ are

$$f_{\mathrm{T}}(X \mid X') = f_{\mathrm{T}}(\emptyset \mid X') \sum_{\theta} \prod_{i:\theta(i)>0} \frac{p_S(x'_i)\, f_{\mathrm{T}}(x_{\theta(i)} \mid x'_i)}{1 - p_S(x'_i)}$$

with

$$f_{\mathrm{T}}(\emptyset \mid X') \triangleq \prod_{x'_i \in X'} \left(1 - p_S(x'_i)\right)$$

and

$$G_{\mathrm{T}}[h \mid X'] = \prod_{x' \in X'} \left(1 - p_S(x') + p_S(x')\, f_{\mathrm{T}}[h \mid x']\right) \tag{5.1}$$

with

$$f_{\mathrm{T}}[h \mid x'] \triangleq \int h(x)\, f_{\mathrm{T}}(x \mid x')\, \mathrm{d}x,$$

where $\theta : \{1, \ldots, n'\} \to \{1, \ldots, n\}$ is the association map similar to (4.2) associating the state $x'_i$ with the state $x_{\theta(i)}$.

*Proof.* The proof for single object transition density $f_T(X \mid X')$ is provided in [Mah07b, Ch. 13.2.3] and the corresponding PGFL $G_T[h \mid X']$ is provided in [Mah07b, Ch. 13.2.6].

Furthermore, recalling that the multi–object model also allows the advent or birth of new objects, the complete multi–object Markov transition model becomes as follows.

**Definition 1.** Let $f_T(X \mid X')$ be the multi–object PDF as described in Proposition 5 and $f_B(X \mid X')$ be the birth PDF then the conditional PDF $f_M(X \mid X')$ and its corresponding PGFL $G_M[h \mid X']$ are

$$\begin{aligned} f_M(X \mid X') &= \sum_{W \subseteq X} \frac{G_B}{\delta W}[0 \mid X'] \, \frac{\delta G_T}{\delta (X \setminus W)}[0 \mid X'] \\ &= \sum_{W \subseteq X} f_B(W \mid X') \, f_T(X \setminus W \mid X') \end{aligned}$$

and

$$G_M[h \mid X'] = G_B[h \mid X'] \, G_T[h \mid X']$$

with

$$G_B[h \mid X'] \triangleq \int h^X \, f_B(X \mid X') \delta X$$

being the PGFL of the multi–object birth PDF.

## 5.3. Multi–Object Birth Models

In case of the birth model, there are basically three models that have been widely employed. Object birth is in general either considered to be Poisson, independent and identically distributed cluster (IIDC) or multi–object multi–Bernoulli (MeMBer) distributed. In the following, the principal equations of the Markov PDFs and PGFLs will be summarized. Unless otherwise specified, objects may be born spontaneously or by spawning from an existent object. Due to the latter, the birth PDFs are considered to be conditionally dependent on the multi–object state.

### 5.3.1. Poisson Birth Model

One of the most widely employed birth models is the Poisson Birth Model. Following the definition of the Poisson random finite set (RFS) in Section 3.1 the corresponding multi–object Markov model equations become as stated in Proposition 6.

**Proposition 6.** Let the birth model be Poisson with average birth rate $\lambda_{\mathrm{B}}(X')$ and single–object birth density $f_{\mathrm{B}}(x \mid X')$, then the multi–object conditional PDF $f_{\mathrm{M}}(X \mid X')$ and PGFL $G_{\mathrm{M}}[h \mid X']$ become

$$f_{\mathrm{M}}(X \mid X') = f_{\mathrm{B}}(X \mid X')\, f_{\mathrm{T}}(\emptyset \mid X') \sum_{\theta} \prod_{i:\theta(i)>0} \frac{p_S(x'_i)\, f_{\mathrm{T}}(x_{\theta(i)} \mid x'_i)}{\left(1 - p_S(x'_i)\right)\, \lambda_{\mathrm{B}}(X')\, f_{\mathrm{B}}(x_{\theta(i)} \mid X')}$$

and

$$G_{\mathrm{M}}[h \mid X'] = G_{\mathrm{B}}[h \mid X'] \prod_{x' \in X'} (1 - p_S(x') + p_S(x')\, f_{\mathrm{T}}[h \mid x'])$$

with

$$G_{\mathrm{B}}[h \mid X'] = e^{\lambda_{\mathrm{B}}(X')(f_{\mathrm{B}}[h|X']-1)}$$

where

$$f_{\mathrm{B}}(X \mid X') = e^{-\lambda_{\mathrm{B}}(X')} \prod_{i=1}^{n} \lambda_{\mathrm{B}}(X')\, f_{\mathrm{B}}(x_i \mid X')$$

$$f_{\mathrm{B}}[h \mid X'] \triangleq \int h(x)\, f_{\mathrm{B}}(x \mid X')\, \mathrm{d}x.$$

*Proof.* The proof follows directly by replacing the birth PDF and PGFL by the corresponding Poisson process as described in Section 3.1. □

### 5.3.2. Independent and Identically Distributed Cluster Birth Model

While the Poisson model is the de facto *standard* model, it has some major drawbacks. Most noticeable is the fact that not only the expected cardinality of the FS of new born objects is equal to the average birth rate $\lambda_{\mathrm{B}}$ but more importantly, the variance of the cardinality is also equal to the birth rate $\lambda_{\mathrm{B}}$. That is why, the uncertainty increases with the amount of possible objects in the monitored area. A generalization of the Poisson birth model is the IIDC birth model which replaces the Poisson cardinality distribution with an arbitrary discrete cardinality distribution $p_{\mathrm{B}}(n)$. In turn the Markov density and its PGFL become as stated in Proposition 7.

**Proposition 7.** Let the RFS of new born objects be modeled as an IIDC RFS as described in Section 3.2 with probability mass function (PMF) $p_{\mathrm{B}}(n \mid X')$ and single–object PDF $f_{\mathrm{B}}(x \mid X')$, then the Markov PDF and PGFL become

$$f_{\mathrm{M}}(X \mid X') = \sum_{W \subseteq X} f_{\mathrm{B}}(W \mid X')\, f_{\mathrm{T}}(X \setminus W \mid X')$$

and

$$G_{\mathrm{M}}[h \mid X'] = G_{\mathrm{B}}[h \mid X'] \prod_{x' \in X'} (1 - p_S(x') + p_S(x')\, f_{\mathrm{T}}[h \mid x'])$$

with

$$G_{\mathrm{B}}[h \mid X'] = \sum_{n \geq 0} p_{\mathrm{B}}(n \mid X')\ (f_{\mathrm{B}}[h \mid X'])^n,$$

where

$$f_{\mathrm{B}}(X \mid X') = |X|!\, p_{\mathrm{B}}(|X| \mid X') \prod_{x \in X} f_{\mathrm{B}}(x \mid X')$$

is the PDF of the birth model/RFS and $f_{\mathrm{T}}(X \mid X')$ is the PDF of the transition as described in Proposition 5.

*Proof.* The proof follows directly by replacing the birth PDF and PGFL by the corresponding IIDC process as described in Section 3.2. □

### 5.3.3. Multi–Bernoulli Birth Model

Besides the Poisson and the IIDC model, which is for example employed in the derivation of the probability hypothesis density (PHD) filter, there are other important birth models. One of those is the multi–Bernoulli birth model. In the Bernoulli model each new born object is described by a Bernoulli RFS. Considering that each new born object's probability of existence $q_{\mathrm{B}_i}$ and its corresponding PDF $f_{\mathrm{B}_i}(X \mid X')$ depends on the current state of objects $X'$, then the corresponding Markov density and PGFL will be as presented in Proposition 8.

**Proposition 8.** Let the birth model be MeMBer with $\nu_{\mathrm{B}}$ Bernoulli birth components $\{(q_{\mathrm{B}i}, f_{\mathrm{B}i})_{i=1}^{\nu_{\mathrm{B}}}\}$ then the PGFL becomes

$$f_{\mathrm{M}}(X \mid X') = \sum_{W \subseteq X} f_{\mathrm{B}}(W \mid X')\ f_{\mathrm{T}}(X \setminus W \mid X')$$

$$G_{\mathrm{M}}[h \mid X'] = G_{\mathrm{B}}[h \mid X'] \prod_{x' \in X'} (1 - p_S(x') + p_S(x') f_{\mathrm{T}}[h \mid x'])$$

with

$$G_{\mathrm{B}}[h \mid X'] = \prod_{i=1}^{\nu_{\mathrm{B}}} (1 - q_{\mathrm{B}_i}(X') + q_{\mathrm{B}_i}(X') f_{\mathrm{B}_i}[h \mid X']),$$

where $f_{\mathrm{T}}(X \mid X')$ is the PDF of the Markov transition including object survival as described in Proposition 5 and

$$f_{\mathrm{B}}(X \mid X') = \prod_{i=1}^{\nu_{\mathrm{B}}} (1 - q_{\mathrm{B},i}(X')) \sum_{1 \leq i_1 \neq \ldots \neq i_{|X|} \leq \nu_{\mathrm{B}}} \prod_{j=1}^{|X|} \frac{q_{\mathrm{B}_{i_j}}(X')\, f_{\mathrm{B}_i}(x_{i_j} \mid X')}{1 - q_{\mathrm{B}_{i_j}}(X')}$$

is the *PDF* of the MeMBer birth RFS and

$$f_{\mathrm{B}}[h \mid X'] \triangleq \int h(x)\, f_{\mathrm{B}}(x \mid X')\mathrm{d}x$$

is the single–object PGFL of the single–object birth PDF $f_{\mathrm{B}}(x \mid X')$.

*Proof.* The proof follows directly by replacing the birth PDF and PGFL by the corresponding Multi–Bernoulli process as described in Section 3.1. □

# Part III.

# Realizations of RFS Filter

# 6. Multi–Object Bayes Filtering

## Contents



## Introduction

In Chapter 4 the state space models for a time varying number of multiple objects have been introduced. Furthermore, likelihood models for detection–type (sec. 4.3) and superpositional (SPS)–type (sec. 4.4.1) sensor models have been presented. Then, in Chapter 5 multi–object transition densities with the according Markov densities and birth models have been summarized. All the established information provide the necessary tools to derive Bayesian filters for estimating the state of multiple objects. But before presenting the principal Bayesian filter equations for SPS–type sensors, the basic multi–object Bayes filter predictor and corrector will be recalled.

## 6.1. Single–Object Bayes Filters

Bayesian filters are a well studied and often employed tool for engineers that allows the estimation of an unknown probability density function (PDF) recursively over time. The unknown PDF is determined using a time series of measurements provided by a sensor, the probabilistic description of the sensor's behavior and a probabilistic model of the state transition over time, denoted as the Markov transition density.

In general, it is assumed that a sensor has collected a sequence of vector valued measurements $z_{1:k} \overset{\text{abbr}}{=} (z_1, \ldots, z_k)$ up to time $k$ where each is carrying information about the vector–valued state of an object $x$ at the same time. Using this information, the Bayes filter provides a possibility to estimate the object state $x$ or better the PDF $f_{k|k}(x)$ of the *uncertain* object state $x$ at time $k$. In general,

the Bayes filter operates in two steps, the prediction and the correction step. Commonly, the prediction step is solely based on the system dynamics described through the so called Markov transition density $f_{\mathrm{M}}(x \mid x')$. Thus, the previous PDF $f_{k-1|k-1}(x \mid z_{1:k-1})$ at time $k-1$ is transitioned to the a priori or predicted PDF $f_{k|k-1}(x \mid z_{1:k-1})$. The correction step uses the current measurement $z_k$ and a sensor model described by its corresponding likelihood $f_k(z_k \mid x)$ to update the knowledge about the uncertain object state $x$ by correcting the predicted PDF which results in the corrected PDF $f_{k|k}(x \mid z_{1:k})$. Hence, given an initial PDF $f_{0|0}(x)$ the filter propagates the PDF over time

$$\begin{aligned} f_{0|0}(x) \rightarrow \quad & f_{1|0}(x) \rightarrow \quad & f_{1|1}(x \mid z_1) \rightarrow \\ \ldots \rightarrow \quad & f_{k|k-1}(x \mid z_{1:k-1}) \rightarrow \quad & f_{k|k}(x \mid z_{1:k}) \rightarrow \ldots \end{aligned}$$

using the Bayes predictor and corrector equations. The principal Bayes filter equations for vector–valued measurements and states are summarized in the following sections.

### 6.1.1. Single–Object Bayes Filter Predictor

As mentioned before, the Bayes filter predictor essentially propagates the given PDF $f_{k-1|k-1}(x \mid z_{1:k-1})$ from time $k-1$ to $k$ given an arbitrary Markov transition density $f_{\mathrm{M}}(x \mid z_{1:k-1})$ resulting in a predicted PDF $f_{k|k-1}(x \mid z_{1:k-1})$. While in general arbitrary, the Markov transition density often describes the underlying system dynamics, such as the principal motion of objects. The general Bayes filter equation is as follows.

**Proposition 9.** Let $z_{1:k-1} \overset{\text{abbr}}{=} (z_1, \ldots, z_{k-1})$ be a sequence of vector–valued measurements collected up to time $k-1$, $f_{\mathrm{M}}(x \mid x')$ be the Markov transition density and $f_{k-1|k-1}(x' \mid z_{1:k-1})$ be the a posteriori PDF, then the Bayes predictor equation for iteration $k$ is

$$f_{k|k-1}(x \mid z_{1:k-1}) = \int f_{\mathrm{M}}(x \mid x')\ f_{k-1|k-1}(x' \mid z_{1:k-1})\, \mathrm{d}x'.$$

*Proof.* The correctness of the Bayes filter predictor equation can be shown by induction. The complete derivation can be found in [TBF06, Sec. 2.4.3].

### 6.1.2. Single–Object Bayes Filter Corrector

If provided with the predicted PDF $f_{k|k-1}(x \mid z_{1:k-1})$, then the Bayes filter corrector uses the information provided by the current measurement $z_k$ at time $k$ to update the current knowledge about the uncertain state $x$. Thus given any likelihood $f_k(z_k \mid x)$ describing the correlation between the measurement $z$ and the state $x$, the corrected PDF $f_{k|k}(x \mid z_{1:k})$ can be inferred.

**Proposition 10.** Let $z_k$ be the measurement and $f_k(z_k \mid x)$ be the measurement likelihood and $f_{k|k-1}(x \mid z_{1:k-1})$ is the predicted PDF at iteration $k$ , then the Bayes corrector equation is

$$f_{k|k}(x \mid z_{1:k}) = \frac{f_k(z_k \mid x)\ f_{k|k-1}(x \mid z_{1:k-1})}{\int f_k(z_k \mid x)\ f_{k|k-1}(x \mid z_{1:k-1})\ \mathrm{d}x}.$$

*Proof.* The correctness of the Bayes filter corrector equation can be shown by induction. The complete derivation can be found in [TBF06, Sec. 2.4.3].

## 6.2. Multi–Object Bayes Filters

An equivalent formulation of the Bayes filter exists for the multi–object scenario. The multi–object Bayes Filter as presented in [Mah07b, Ch. 14] assumes that a sensor has collected a sequence of finite set (FS)–valued measurements $Z_{1:k} \overset{\text{abbr}}{=} (Z_1, \ldots, Z_k)$, where each FS–valued measurement at time $k$ is comprised out of $m_k$ vector–valued measurements $Z_k = \{z_{(k,1)}, \ldots, z_{(k,m_k)}\}$. It is assumed that each FS–valued measurement $Z_k$ carries information about a FS–valued state $X$ at time $k$, which describes the current state of $n_k$ vector–valued states $X_k = \{x_{(k,1)}, \ldots, x_{(k,n_k)}\}$.

Similar to the single–object case the multi–object Bayes filter is a two step procedure. But, instead of predicting and correcting the single–object PDF $f_{k|k}(x \mid z_{1:k})$, it predicts and corrects the multi–object PDF $f_{k|k}(X \mid Z_{1:k})$ . Thus, given an initial multi–object PDF $f_{0|0}(X)$ the multi–object Bayes filter provides a sequence of multi–object PDFs over time

$$\begin{aligned} f_{0|0}(X) \rightarrow \quad & f_{1|0}(X) \rightarrow \quad & f_{1|1}(X \mid Z_1) \rightarrow \\ \ldots \rightarrow \quad & f_{k|k-1}(X \mid Z_{1:k-1}) \rightarrow \quad & f_{k|k}(X \mid Z_{1:k}) \rightarrow \ldots \end{aligned}$$

The principal Bayes filter equations for the prediction and correction are presented in the following. For an in–depth explanation one is referred to [Mah07b, Ch. 14].

### 6.2.1. Multi–Object Bayes Filter Predictor

Given the definition of the set–integral in (2.1), the multi–object Bayes filter predictor equation looks quite similar to the ordinary single–object Bayes filter equations. As described in Chapter 5, the multi–object Markov density $f_{k|k-1}(X \mid X')$ does not only describe the pure transition of the state but also describes the survival and birth of objects. Therefore, given any of the proposed Markov models and densities presented in Section 5.2, the PDF form of multi–object Bayes predictor equation becomes as stated in the following.

**Proposition 11.** Let $Z_{1:k-1} \overset{\text{abbr}}{=} (Z_1, \ldots, Z_{k-1})$ be a time–sequence of FS–valued measurement up to time $k-1$, $f_{\mathrm{M}}(X \mid X')$ be a multi–object Markov density as de-

scribed in Chapter 5 and the a priori PDF $f_{k-1|k-1}(X \mid Z_{1:k-1})$, then the multi–object Bayes predictor becomes

$$f_{k|k-1}(X \mid Z_{1:k-1}) = \int f_{\mathrm{M}}(X \mid X')\ f_{k-1|k-1}(X' \mid Z_{1:k-1})\,\delta X'.$$

*Proof.* The multi–object Bayes filter predictor equation is a direct result of the Bayes' rule as shown in [Mah07b, Sec. 14.3].

While these equations can be used to derive more specific multi–object Bayes filters, it is often useful to work with the probability generating functional (PGFL) form of the Bayes filter equations. When consequently applying the principles of multi–object calculus presented in Chapter 2, then the PGFL variant for the predictor becomes as stated in the following.

**Proposition 12.** Let the multi–object Bayes filter predictor equation be as proposed in Proposition 11, then its corresponding PGFL form is

$$G_{k|k-1}[h \mid Z_{1:k-1}] = \int G_{\mathrm{M}}[h \mid X']\ f_{k-1|k-1}(X' \mid Z_{1:k-1})\,\delta X'$$

where

$$G_{\mathrm{M}}[h \mid X'] = \int h^X\ f_{\mathrm{M}}(X \mid X')\,\delta X$$

is the PGFL of the multi–object Markov density with respect to $X$ and $h^X \triangleq \prod_{x \in X} h(x)$ is the set–power of test–function $h(x)$ where $h^{\emptyset} \triangleq 1$.

*Proof.* The result follows directly by transforming the multi–object Bayes filter predictor from Proposition 11 into its PGFL form. The derivation can be found in [Mah07b, Sec. 14.8.1].

Starting from the PGFL form of the random finite set (RFS) Bayes filter predictor, the PGFL form of the RFS Bayes filter for the standard Markov model as described in Section 5.2 can easily be established.

*Remark* 1 (Multi–Object Bayes Filter Predictor for the Standard Markov Model). Considering that the PGFL of the Markov density is as described in Definition 1, then the RFS Bayes filter predictor equation becomes

$$G_{k|k-1}[h \mid Z_{1:k-1}] = \int G_{\mathrm{B}}[h \mid X']\ G_{\mathrm{T}}[h \mid X']\ f_{k-1|k-1}(X' \mid Z_{1:k-1})\ \delta X'$$

with $G_{\mathrm{T}}[h \mid X']$ being the PGFL describing the multi–object transition as stated in (5.1) and $G_{\mathrm{B}}[h \mid X']$ being the PGFL of an arbitrary multi–object birth distribution.

### 6.2.2. Multi–Object Bayes Filter Corrector

The multi–object Bayes filter corrector updates the knowledge of an uncertain FS–valued state $X$ using the information provided by the FS–valued measurement $Z_k$ at time $k$. Thus, employing any multi–object measurement likelihood $f_k(Z_k \mid X)$ as proposed in Chapter 4, the corrected multi–object PDF $f_{k|k}(X \mid Z_{1:k})$ is inferred as described in the following.

**Proposition 13.** Let $f_k(Z_k \mid X)$ be a multi–object likelihood as described in Chapter 4 and $Z_k$ be the FS–valued measurement at time $k$, then the multi–object equivalent of the Bayes filter corrector becomes

$$f_{k|k}(X \mid Z_{1:k}) = \frac{f_k(Z_k \mid X)\ f_{k|k-1}(X \mid Z_{1:k-1})}{\int f_k(Z_k \mid X)\ f_{k|k-1}(X \mid Z_{1:k-1})\ \delta X}. \tag{6.1}$$

*Proof.* The multi–object Bayes filter corrector equation is a direct result of the Bayes' rule as shown in [Mah07b, Sec. 14.4].

It should be noted that while the single– and multi–object Bayes filter equations might look quite similar at a first glance, much of its complexity is hidden behind the set–integral as described in (2.1).

Even in the corrector case, it is often useful to work with the PGFL form of the Bayes filter corrector as described subsequently.

**Proposition 14.** Be the multi–object Bayes filter corrector as stated in Proposition 13, then the PGFL form of the equations is

$$G_{k|k}[h \mid Z_{1:k}] = \frac{F_{Z_k}[h]}{F_{Z_k}[1]} \tag{6.2}$$

with

$$\begin{aligned} F_{Z_k}[h] &= \left(\frac{\delta F}{\delta Z_K}\right)[0,h] \\ &= \int h^X\ f_k(Z_k \mid X)\ f_{k|k-1}(X \mid Z_{1:k-1})\ \delta X \end{aligned}$$

being the functional derivative with respect to the measurement set of

$$F[g,h] = \int h^X\ G_k[g \mid X]\ f_{k|k-1}(X \mid Z_{1:k-1})\ \delta X$$

where

$$G_k[g \mid X] = \int g^Z\ f_k(Z \mid X)\ \delta Z$$

is the PGFL of the multi–object likelihood $f_k(Z_k \mid X)$ and $g^Z \triangleq \prod_{z \in Z} g(z)$ with $g^{\emptyset} \triangleq 1$.

*Proof.* In [Mah07b, App. G. 25] it is proven that the PGFL form of the Bayes filter corrector is as proposed in (6.2). □

Taking a closer look at Proposition 14 reveals that it is not possible to obtain the corrected PGFL $G_{k|k}[h \mid Z_{1:k}]$ from the PGFL of the predicted $G_{k|k-1}[h \mid Z_{1:k-1}]$ directly but from the PDF $f_{k|k-1}(X \mid Z_{1:k-1})$.

Specialized versions of the multi–object Bayes filter can easily be derived for the detection– and SPS–type measurement models as described in Chapter 4.

**Proposition 15** (Detection–Type Multi–Object Bayes Filter Corrector)**.** Given the detection–type likelihood $f(Z \mid X)$ as described in Proposition 1 with Poisson clutter, then the multi–object Bayes corrector becomes

$$f_{k|k}(X \mid Z_{1:k}) = \frac{f(\emptyset \mid X) \sum_{\theta} \prod_{i:\theta(i)>0} \frac{p_D(x_i)\, f(z_{\theta(i)}|x_i)}{(1-p_D(x_i))\, \lambda_C f_C(z_{\theta(i)})}\, f_{k|k-1}(X \mid Z_{1:k-1})}{\int f(\emptyset \mid X) \sum_{\theta} \prod_{i:\theta(i)>0} \frac{p_D(x_i)\, f(z_{\theta(i)}|x_i)}{(1-p_D(x_i))\, \lambda_C f_C(z_{\theta(i)})}\, f_{k|k-1}(X \mid Z_{1:k-1})\, \delta X}$$

with the PDF that all objects are missed

$$f(\emptyset \mid X) \triangleq \prod_{x_i \in X} (1 - p_D(x_i)).$$

Furthermore, the corresponding PGFL is

$$G_{k|k}[h \mid Z_{1:k}] = \frac{F_{Z_k}[h]}{F_{Z_k}[1]}$$

where the nominator and denominator are defined as

$$F_{Z_k}[h] \triangleq \frac{\delta F}{\delta Z_k}[0, h]$$

and

$$F[g, h] \triangleq \mathrm{e}^{\lambda_C\, (f_C[g]-1)} \int \prod_{x \in X} (1 - p_D(x) + p_D(x) f[g \mid x])\, f_{k|k-1}(X \mid Z_{1:k-1})\, \delta X.$$

Also, notice that $\frac{\delta F}{\delta Z_k}[0, h]$ is the evaluation of the set–derivative $\frac{\delta F[g_h]}{\delta Z_k}$ at $g(z) = 0$.

*Proof.* The result follows directly by inserting the definition of the detection–type measurement likelihood from Proposition 1 into (6.1) or its PGFL from Proposition 2 into (6.2) respectively. □

Similar results can be established for the SPS–type measurement model.

**Proposition 16** (Superposition–Type Multi–Object Bayes Filter Corrector)**.** Given the SPS measurement likelihood $f(z \mid X)$ as described in Proposition 4 then the multi–object Bayes corrector becomes

$$f_{k|k}(X \mid Z_{1:k}) = \frac{\left(f_w * \Psi^{(*X)}\right)(z_k)\ f_{k|k-1}(X \mid Z_{1:k-1})}{\int \left(f_w * \Psi^{(*X)}\right)(z_k)\ f_{k|k-1}(X \mid Z_{1:k-1})\ \delta X}$$

with PGFL

$$G_{k|k}[h \mid Z_{1:k}] = \frac{\int h^X\ \left(f_w * \Psi^{(*X)}\right)(z_k)\ f_{k|k-1}(X \mid Z_{1:k-1})\ \delta X}{\int \left(f_w * \Psi^{(*X)}\right)(z_k)\ f_{k|k-1}(X \mid Z_{1:k-1})\ \delta X}$$

with the convolution set–power being defined as

$$\Psi^{(*X)}(z) = \begin{cases} \delta_0(z) & \text{if } X = \emptyset \\ (\Psi_1 * \ldots * \Psi_n)(z) & \text{else} \end{cases}$$

where $X = \{x_1, \ldots, x_n\}$ and $\Psi_i(z \mid x_i)$ is defined in (4.12).

*Proof.* The result follows directly by inserting the definition of the SPS measurement likelihood from (4.11) in (6.1) and (6.2). □

# 7. Multi–Bernoulli Realizations

## Contents



## Introduction

In Chapter 6 the principle equations for the multi–object Bayes filter were introduced. In theory, these filters allow estimating the state of multiple objects recursively, in practice a specific probability density function (PDF) function needs to be chosen to derive a usable and computationally tractable filter.

If taking a look at typical single object Bayesian filter realizations, such a specific PDF could either be a Gaussian PDF or an arbitrary PDF approximated by a set of weighted samples, often also called particles. Under the assumption that the single–object Markov transition function $\tau(x)$ and single–object measurement function $\eta(x)$ are both linear with Gaussian transition density and Gaussian measurement likelihood, then the resulting filter is the well–known Kalman filter ([TBF06, Ch. 3]). In the latter case, when the PDF is chosen to be approximated by a finite number of samples/particles, the result would be the Particle filter ([TBF06, Ch. 4]).

In multi–object Bayesian filtering the PDF could in theory be any valid multi–object PDF. However, in practice the Poisson, the independent and identically distributed cluster (IIDC) or the multi–Bernoulli PDFs have been shown to lead to viable multi–object Bayes filter realizations. For example choosing the PDF to be either Poisson (sec. 3.1) or IIDC (sec. 3.1), then the resulting Bayes filter allows the derivation of the prominent class of probability hypothesis density (PHD)[Mah03] and cardinalized probability hypothesis density (CPHD) [Mah07a] filters, which have been employed successfully in many multi–object filtering scenarios. Moreover, choosing the PDF to be multi–Bernoulli as in Section 3.4 leads

to the derivation of the multi–object multi–Bernoulli (MeMBer) and cardinality balanced multi–object multi–Bernoulli (CB-MeMBer) class of filters.

In general, choosing the PDF in this way provides an efficient way to propagate the PDF over time, since only the Bernoulli parameters have to be propagated. That is why, given an initial set of Bernoulli parameters $\left\{\left(q_{0|0\,i}, s_{0|0\,i}(x)\right)\right\}_{i=1}^{\nu_{0|0}}$ the MeMBer Bayes filters provide a sequence of Bernoulli parameters over time.

$$\begin{aligned}\left\{\left(q_{0|0\,i}, s_{0|0\,i}(x)\right)\right\}_{i=1}^{\nu_{0|0}} \rightarrow \quad & \left\{\left(q_{1|0\,i}, s_{1|0\,i}(x)\right)\right\}_{i=1}^{\nu_{1|0}} \rightarrow \quad \left\{\left(q_{1|1\,i}, s_{1|1\,i}(x)\right)\right\}_{i=1}^{\nu_{1|1}} \rightarrow \\ \ldots \rightarrow \quad & \left\{\left(q_{k|k-1\,i}, s_{k|k-1\,i}(x)\right)\right\}_{i=1}^{\nu_{k|k-1}} \rightarrow \quad \left\{\left(q_{k|k\,i}, s_{k|k\,i}(x)\right)\right\}_{i=1}^{\nu_{k|k}} \rightarrow \ldots\end{aligned}$$

While all of the mentioned classes of multi–object Bayesian filters have been studied in recent years, they are mostly based on the assumption that the sensors provide measurements according to the detection–type model as described in Section 4.3. However, in this chapter the focus is on a multi–Bernoulli Bayes filter realization for superpositional (SPS)–type sensors as described in Section 4.4. The CB-MeMBer filter equations for detection–type filters will be recalled in Section 7.1. Then, the principal equation for multi–Bernoulli filters will be derived in Section 7.2.

## 7.1. MeMBer Filters for Detection–Type Sensors

In recent years, Multi–Bernoulli filters have been shown to be a valid asset when estimating the state of a time–varying number of objects when the sensors provide detection–type measurements. The most prominent multi–Bernoulli filter realization is the CB-MeMBer filter which was proposed by Vo, Vo, and Cantoni in [VVC09] as an improvement to the previously derived MeMBer filter by Mahler in [Mah07b]. Since then the CB-MeMBer filter and its derivates have been applied to various state estimation problems most typically with detection–type sensors providing some kind of distance and angular measurements towards the objects as for example explained in [Ris+13], but also in visual tracking scenarios as described in [HVV11].

Since some of the results from the detection–type CB-MeMBer will aid in the understanding of the SPS–type MeMBer filter realization and its derivatives, the main CB-MeMBer equations will be summarized in the upcoming sections.

### 7.1.1. MeMBer Filter Predictor

In order to derive the CB-MeMBer predictor equations the following assumptions are made:

1. The initial random finite set (RFS) $X_0$ is a multi–Bernoulli RFS as described in Section 3.4.

2. The Markov transition model is the standard model as described in Chapter 5 with single–object probability of survival $p_S(x)$ and Markov density $f_{\mathrm{M}}(x \mid x')$,
3. New objects are presumed to be generated due to the multi–Bernoulli Birth model as described in 5.3.3.

Considering the aforementioned assumptions the CB-MeMBer predictor equations are as follows.

**Proposition 17.** Given the posterior multi–object distribution at time $k-1$ is multi–Bernoulli with set of $\nu \overset{\text{abbr}}{=} \nu_{k-1|k-1}$ parameters

$$\{(q_i, s_i(x))_{i=1}^{\nu}\},$$

where $q \overset{\text{abbr}}{=} q_{k-1|k-1}$ is the probability that an object exists at time $k-1$ and $s(x) \overset{\text{abbr}}{=} s_{k-1|k-1}(x)$ is its corresponding spatial PDF. Furthermore, let

$$\left\{\left(q_{\mathrm{B}_i}, s_{\mathrm{B}_i}(x)\right)_{i=1}^{\nu_{\mathrm{B}}}\right\}$$

be the set of Bernoulli parameters for the new born objects, where the $i$–th new object exists with probability $q_{\mathrm{B}_i}$ and PDF $s_{\mathrm{B}_i}(x)$.

Also, let

$$\left\{\left(q_{\mathrm{T}_i}, s_{\mathrm{T}_i}(x)\right)_{i=1}^{\nu}\right\}$$

be the set of Bernoulli parameters for the transitioned objects, where the $i$–th transitioned object exists with probability $q_{\mathrm{T}_i}$ and PDF $s_{\mathrm{T}_i}(x)$.

Then, the $\nu' = \nu + \nu_{\mathrm{B}}$ parameters $\left(q_i', s_i'(x)\right)_{i=1}^{\nu'}$ with $q' \overset{\text{abbr}}{=} q_{k|k-1}$, $s'(x) \overset{\text{abbr}}{=} s_{k|k-1}(x)$ and $\nu' \overset{\text{abbr}}{=} \nu_{k|k-1}$ become

$$\left\{\left(q_i', s_i'(x)\right)_{i=1}^{\nu'}\right\} = \left\{\left(q_{\mathrm{T}_i}, s_{\mathrm{T}_i}(x)\right)_{i=1}^{\nu}\right\} \cup \left\{\left(q_{\mathrm{B}_i}, s_{\mathrm{B}_i}(x)\right)_{i=1}^{\nu_{\mathrm{B}}}\right\}$$

after prediction.

Presuming that $p_s(x)$ is the single–object probability of survival and $f_{\mathrm{M}}(x \mid x')$ is an arbitrary single–object Markov density, then the transitioned parameters $\left(q_{\mathrm{T}_i}, s_{\mathrm{T}_i}(x)\right)$ are determined by

$$q_{\mathrm{T}_i} = q_i \; s_i[p_S]$$

and

$$s_{\mathrm{T}_i}(x) = \frac{s_i[p_S \; f_{\mathrm{M}}]}{s_i[p_S]}$$

with

$$s_i[h] \triangleq \int h(x) \, s_i(x) \, \mathrm{d}x$$

being the probability generating functional (PGFL) of the single–object PDF $s_i(x)$.

*Proof.* The results of the proposition are derived directly from the PGFL form of the multi–object Bayes filter predictor as described in Proposition 12. A detailed proof is provided by Mahler in [Mah07b, Sec. 17.4.1].

Having a look at Proposition 17 it is noticeable that under the aforementioned assumption the multi–object Bayes prediction decomposes into multiple parallel single–object predictions and as such has a linear computational complexity $\mathcal{O}(\nu)$ in the number of Bernoulli components.

### 7.1.2. Cardinality Balanced MeMBer Filter Corrector

In order to derive the CB-MeMBer corrector equations the following is assumed:

1. The predicted PDF is the PDF of a multi–Bernoulli RFS with component probabilities of existence $q_i'$ and component PDF $s_i'(x)$.

2. The sensor model is the detection–type model as described in Section 4.3 with single–measurement likelihood $f(z \mid x)$ and single–object probability of detection $p_D(x)$.

3. False measurements are modeled by Poisson clutter with rate $\lambda_C$ and single–measurement clutter PDF $f_C(z)$.

4. No two measurements $z_i, z_j$ will be close to or originate from the same object such that the joint likelihood $f(z_i \mid x)\, f(z_j \mid x) \approx 0$.

Considering this, the MeMBer predictor equations become as follows.

**Proposition 18.** Let $\left\{\left(q_i', s_i'(x)\right)_{i=1}^{\nu'}\right\}$ be the predicted Bernoulli components and $Z = \{z_1, \ldots, z_m\}$ be the set of $m$ collected measurements at time $k$ where $Z \overset{\text{abbr}}{=} Z_k$, $z \overset{\text{abbr}}{=} z_k$ and $m \overset{\text{abbr}}{=} m_k$. Then, the corrected Bernoulli parameters $\left\{(q_i, s_i(x))_{i=1}^{\nu}\right\}$ with $q \overset{\text{abbr}}{=} q_{k|k}$, $s(x) \overset{\text{abbr}}{=} s_{k|k}(x)$ and $\nu \overset{\text{abbr}}{=} \nu_{k|k}$ are the union of the missed $\left\{\left(q_{m_i}, s_{m_i}(x)\right)_{i=1}^{\nu'}\right\}$ and measurement induced set of parameters $\{(q_d(z_i), s_d(x \mid z_i))_{i=1}^{m}\}$ being equal to

$$\{(q_i, s_i(x))_{i=1}^{\nu}\} = \left\{\left(q_{m_i}, s_{m_i}(x)\right)_{i=1}^{\nu'}\right\} \cup \{(q_d(z_i), s_d(x \mid z_i))_{i=1}^{m}\}.$$

The individual missed components are determined by

$$q_{m_i} = \frac{1 - s_i'[p_D]}{1 - q_i'\, s_i'[p_D]}\, q_i'$$

$$s_{m_i}(x) = \frac{1 - p_D(x)}{1 - s_i'[p_D]}\, s_i'(x)$$

and with $f_z(x) \stackrel{\text{abbr}}{=} f_k(z \mid x)$ the measurement induced components are

$$q_d(z_i) \approx \frac{\sum_{j=1}^{\nu'} \frac{(1-q_j')}{\left(1-q_j' s_j'[p_D]\right)} \frac{q_j' s_j\left[p_D\, f_{z_i}\right]}{\left(1-q_j' s_j'[p_D]\right)}}{\lambda_C f_C(z_i) + \sum_{j=1}^{\nu'} \frac{q_j' s_j'\left[p_D\, f_{z_i}\right]}{1-q_j' s_j'[p_D]}}$$

$$s_d(x \mid z_i) \approx \frac{\sum_{j=1}^{\nu'} \frac{q_j'}{1-q_j'}\, f_{z_i}(x)\, s_j'(x)}{\sum_{j=1}^{\nu'} \frac{q_j'}{1-q_j'}\, s_j'\left[f_{z_i}\right]}$$

where $s'[h] \triangleq \int h(x)\, s'(x)\, \mathrm{d}x$.

*Proof.* The original MeMBer filter corrector equations are provided by Mahler in [Mah07b, Sec. 17.4.2]. Starting with the PGFL form of the multi–object Bayes filter corrector equation, Mahler applies several approximations to derive valid Bernoulli parameters. However, it is shown by Vo, Vo, and Cantoni in [VVC09] that the resulting filter equations lead to a bias in the cardinality estimate. Hence, some modifications to the original member corrector equation are employed in [VVC09] to derive the CB-MeMBer filter equations as proposed in Proposition 18.

Note that the complexity of the CB-MeMBer corrector as described in Proposition 18 is quadratic in the number of Bernoulli components $\nu'$ and measurements $m$ with $\mathcal{O}((m+1)\,\nu')$ which for a multi–object Bayes filter is rather efficient. It should be noted that while the MeMBer predictor equations presented in Proposition 17 resemble the full multi–object Bayes filter prediction equations, the corrector equations do not.

In fact, without going into too much detail the CB-MeMBer corrector approximation is based on the following approximation to the PGFL (see [Mah07b, Sec. 17.4.2])

$$G[h \mid Z_{1:k}] = \prod_{i=1}^{\nu'} G_{m_i}[h] \prod_{z \in Z_k} G_d[h \mid z]$$

where $G_{m_i}[h]$ is the PGFL corresponding to the $i$-th missed detections and $G_d[h \mid z]$ is the PGFL corresponding to a single measurement $z$. Note, that while it can be shown that the first product is a multi–Bernoulli, the second is not. In order to be able to write the second term as a product it is assumed that clutter is not too dense, in the sense that it is unlikely that two measurements will be near the same track (see [Mah07b, pp. 679]). As a result, according to [VVC07] each factor in the second product might not even be a PGFL.

Since $G_d[h \mid z]$ is no Bernoulli, another approximation is introduced in order to find a Bernoulli approximation to $G_d[h \mid z]$. Mahler's first proposal in [Mah07b,

Sec. 17.4.2] lead to the well–known MeMBer corrector approximations which were later shown to lead to a bias in the posterior cardinality. Therefore, Vo, Vo, and Cantoni proposed the so called CB-MeMBer in [VVC07]. The approximation is chosen in the way that the Bernoulli approximation has the same PHD as that of the original PGFL. From that fact, approximate Bernoulli components are derived. However, while the probability of existence can be derived without further approximation, the posterior PDF is only valid if the probability of detection is $p_D \approx 1$. Note, that for smaller $p_D < 1$ the CB-MeMBer corrector is generally overconfident since the approximate PHD will always be greater then the true PHD.

### 7.1.3. Labeled MeMBer Filter Corrector

Another approach to multi–Bernoulli filters is presented in [VV13] denoted as labeled multi–object multi–Bernoulli (L-MeMBer) filter, which in contrast to the CB-MeMBer filter does not need the aforementioned approximations. Instead of relying on multi–Bernoulli RFSs the authors introduced the notion of labeled multi–Bernoulli RFSs. Without going into too much detail, labeled multi–Bernoulli RFSs are basically multi–Bernoulli RFSs augmented with track–labels. On this basis, the authors where able to show that a labeled multi–Bernoulli filter can be derived which given a labeled multi–Bernoulli prior, its posterior will also be a labeled multi–Bernoulli after the correction step. Further information on L-MeMBer filters can be found in [VV13] or [Reu+14].

## 7.2. MeMBer Filters for Superposition–Type Sensors

After recalling the principal equations for the detection–type MeMBer filter in Section 7.1, this section will provide the principal equations for SPS–type MeMBer filters. Since the underlying measurement model does not influence the Bayes prediction (see Proposition 17), the focus will be solely on the Bayes corrector equations.

In the following sections, the principal corrector equations for the SPS–type MeMBer filter will be provided (sec. 7.2.1), including two alternative approximations required to derive useful Bernoulli parameters (sec. 7.2.2 and 7.2.3).

### 7.2.1. Exact $\Sigma$–MeMBer Filter

In Section 6.2.2 the principal predictor and corrector equations for the MeMBer filter for detection–type sensors were presented. In this section, the MeMBer filter corrector equations for SPS–type sensors will be provided. In the following, these class of filters will be denoted as superpositional multi–object multi–Bernoulli ($\Sigma$–MeMBer) as denoted by Mahler in [Mah14].

#### 7.2.1.1. Exact $\Sigma$–MeMBer Corrector

In the following paragraphs, the principal equations for the multi–Bernoulli for SPS–type will be provided.

The $\Sigma$–MeMBer corrector equations are based on the following assumptions:

1. The sensor model is assumed to be a SPS–type sensor model according to Proposition 4 with SPS likelihood $f(Z \mid X)$ with $Z = \{z\}$ being a single SPS measurement.

2. The individual signals of distinct objects may be missed or not received by the sensor with single–object probability of visibility $p_V(x)$.

3. The predicted distribution is a multi–Bernoulli distribution as described in Section 3.4 with probabilities of existence $q_i'$ and corresponding single–object densities $s_i'(x)$.

Considering the aforementioned assumption, the full Bayes filter corrector equations in its PGFL form are as follows.

**Theorem 1.** Presume that $\left\{\left(q_i', s_i'(x)\right)_{i=1}^{\nu'}\right\}$ are the predicted Bernoulli components at time $k$ where $\nu' \overset{\text{abbr}}{=} \nu'_{k|k-1}$, $q' \overset{\text{abbr}}{=} q'_{k|k-1}$ and $s'(x) \overset{\text{abbr}}{=} s'_{k|k-1}(x)$. Let $Z_k = \{z_k\}$ be the finite set (FS) of the singleton SPS–type measurement $z_k$, $f_w(z)$ be the PDF of the measurement noise and $p_V(x)$ be the single–object probability of visibility, then the full $\Sigma$–MeMBer corrector PGFL $G[h \mid Z_{1:k}] \overset{\text{abbr}}{=} G_{k|k}[h \mid Z_{1:k}]$ is

$$G[h \mid Z_{1:k}] = \frac{(f_w * \zeta_1[h] * \ldots * \zeta_{\nu'}[h])(z_k)}{(f_w * \zeta_1[1] * \ldots * \zeta_{\nu'}[1])(z_k)}$$

with

$$\zeta_i[h](z) \triangleq \left(1 - q_i' + q_i'\, s_i'[h\,(1 - p_V)]\right)\delta_0(z) + q_i' s_i'\left[h\; p_V\; \delta_\eta\right](z)$$

and

$$\zeta_i[1](z) \triangleq \left(1 - q_i'\, s_i'[p_V]\right)\delta_0(z) + q_i' s_i'\left[p_V\; \delta_\eta\right](z)$$

where $\delta_\eta \overset{\text{abbr}}{=} \delta_{\eta(x)}(z)$ is the Dirac delta function, $(f_1 * f_2)(z)$ is the convolution of functions $f_1(z)$, $f_2(z)$ and

$$s_i'[h\,(1 - p_V)] = \int h(x)\; (1 - p_V(x))\; s_i'(x)\; \mathrm{d}x$$

$$s_i'\left[h\; p_V\; \delta_\eta\right] = \int h(x)\; p_V(x)\; \delta_{\eta(x)}(z)\; s_i'(x)\; \mathrm{d}x.$$

*Proof.* See Section 7.2.1.2.

While the equation presented in Theorem 1 is the exact PGFL of the Bayes filter corrector, it should be noted that it is *not* the PGFL of a multi–Bernoulli RFS. Consequently, the corrected distribution is *not* conjugate to the predicted distribution. In fact, the corresponding PDF of the resulting distribution is as follows.

**Theorem 2.** Let the PGFL be as described in Proposition 14, then the corresponding PDF $f(X \mid Z_{1:k}) \overset{\text{abbr}}{=} f_{k|k}(X \mid Z_{1:k})$ is

$$\begin{aligned} f_{k|k}(X \mid Z_{1:k}) &= \frac{\prod_{j=1}^{\nu'}(1-q_i') \sum_{1\leq i_1\neq\ldots\neq i_n\leq\nu'} \left(f_w * \xi_{i_1} * \ldots * \xi_{i_n}\right)(z_k)}{(f_w * \zeta_1[1] * \ldots * \zeta_{\nu'}[1])(z_k)} \\ &\equiv \frac{\sum_{1\leq i_1\neq\ldots\neq i_n\leq\nu'} \left(f_w * \xi_{i_1} * \ldots * \xi_{i_n}\right)(z_k)}{\sum_{n=0}^{\nu'} \frac{1}{n!} \sum_{1\leq i_1\neq\ldots\neq i_n\leq\nu'} \left(f_w * \xi_{i_1}[1] * \ldots * \xi_{i_n}[1]\right)(z_k)} \end{aligned}$$

with

$$\xi_{i_j}(z \mid x_j) = \frac{q_{i_j}' \, \Psi_j(z \mid x_j) \, s_{i_j}'(x_j)}{1 - q_{i_j}'}$$

where

$$\Psi_j\left(z \mid x_j\right) = \left(1 - p_V(x_j)\right)\delta_0(z) + p_V(x_j) \, \delta_{\eta(x_j)}(z)$$

as defined in (4.12) in Proposition 4 and

$$\xi_{i_j}[1](z \mid x_j) = \frac{q_{i_j}' \, s_{i_j}'[\Psi_j](z)}{1 - q_{i_j}'}.$$

*Proof.* See Section 7.2.1.2.

There are some important things to notice. As mentioned before, the resulting corrected distribution is *not* a multi–Bernoulli distribution. Additionally, it also seems that the resulting multi–object distribution is no longer independently distributed and as such no longer composed out of a collection of single–object distributions. Therefore, it is not obvious how to derive approximate multi–Bernoulli parameters from the given PGFL or PDF. While this seems to be a dead–end, the results from this section will prove to be a valid starting point for the derivation of valid approximate multi–Bernoulli filters.

#### 7.2.1.2. Derivation of the exact $\Sigma$–MeMBer Corrector

In this section, the results of the previous section will be proven. First, the PGFL form of the $\Sigma$–MeMBer filter corrector as presented in Theorem 1 will be proven and secondly the PDF form of the same will be shown to be as stated in Theorem 2.

In order to proof Theorems 1 and 2, the following results are used. Let $f_k(z_k \mid X)$ be the SPS–type likelihood as proposed in Proposition 4 and $f_{k|k-1}(X \mid Z_{1:k-1})$ be the PDF of a multi–Bernoulli RFS as stated in Section 3.4, then $F_{z_k}(X)$ becomes

$$\begin{aligned} F_{z_k}(X) &= f_k(z_k \mid X)\ f_{k|k-1}(X \mid Z_{1:k-1}) \\ &= \begin{cases} \prod_{i=1}^{\nu'}(1-q_i')\, f_w(z_k) & \text{if } |X| = 0 \\ (f_w * \Psi_1 * \ldots * \Psi_n)(z_k) \prod_{i=1}^{\nu'}(1-q_i') \sum_{1\le i_1\neq\ldots\neq i_n\le\nu'} \prod_{j=1}^{n} \frac{q_{i_j}' s_{i_j}'(x_j)}{1-q_{i_j}'} & \text{if } 1 \le |X| \le \nu' \\ 0 & \text{if } |X| > \nu' \end{cases} \end{aligned}$$

where $X = \{x_1, \ldots, x_n\}$. Consequently, it follows

$$\begin{aligned} F_{z_k}(X) &= \prod_{i=1}^{\nu'}(1-q_i') \sum_{1\le i_1\neq\ldots\neq i_n\le\nu'} (f_w * \Psi_1 * \ldots * \Psi_n)(z_k) \prod_{j=1}^{n} \frac{q_{i_j}'\, s_{i_j}'(x_j)}{1-q_{i_j}'} \\ &= \prod_{i=1}^{\nu'}(1-q_i') \sum_{1\le i_1\neq\ldots\neq i_n\le\nu'} \left( f_w * \frac{q_{i_1}'\, \Psi_1\, s_{i_1}'(x_1)}{1-q_{i_1}'} * \ldots * \frac{q_{i_n}'\, \Psi_n\, s_{i_n}'(x_n)}{1-q_{i_n}'} \right)(z_k) \\ &= \prod_{i=1}^{\nu'}(1-q_i') \sum_{1\le i_1\neq\ldots\neq i_n\le\nu'} \left( f_w * \xi_{i_1} * \ldots * \xi_{i_n} \right)(z_k) \end{aligned} \tag{7.1}$$

where

$$\xi_i\left(z \mid x_j\right) = \frac{q_i'\, \Psi_j(z \mid x_j)\, s_i'(x_j)}{1-q_i'}.$$

*Proof of Theorem 1.* From Proposition 14 it is known that

$$G[h \mid Z_{1:k}] = \frac{F_{Z_k}[h]}{F_{Z_k}[1]}.$$

where the nominator and denominator are defined as

$$F_{Z_k}[h] = \int h^X\ f_k(Z_k \mid X)\ f_{k|k-1}(X \mid Z_{1:k-1})\ \delta X.$$

In the SPS–type measurement case the measurement FS is essentially a singleton with $Z_k = \{z_k\}$, and therefore the nominator $F_{z_k}[k] \overset{\text{abbr}}{=} F_{\{z_k\}}[h]$ becomes

$$F_{z_k}[h] = \int h^X\ F_{z_k}(X)\ \delta X.$$

Thus, after expanding the set–integral $F_{z_k}[h]$ becomes

$$F_{z_k}[h] = F_{z_k}(\emptyset) + \sum_{n=1}^{\infty} \frac{1}{n!} \int_{x_1} \dots \int_{x_n} h(x_1) \cdots h(x_n)\, F_{z_k}(\{x_1, \dots, x_n\})\, \mathrm{d}x_1 \cdots \mathrm{d}x_n.$$

In turn, inserting the results of (7.1) and acknowledging that $F_{z_k}(X) = 0$ if $|X| > \nu'$ then it follows that

$$\begin{aligned}
F_{z_k}[h] &= F_{z_k}(\emptyset) + \prod_{j=1}^{\nu'}(1 - q_j') \sum_{n=1}^{\nu'} \frac{1}{n!} \sum_{1 \le i_1 \ne \dots \ne i_n \le \nu'} \\
&\qquad \left( f_w * \int_{x_1} \dots \int_{x_n} \left( h(x_1)\, \xi_{i_1} \right) * \dots * \left( h(x_n)\, \xi_{i_n} \right) \mathrm{d}x_1 \cdots \mathrm{d}x_n \right)(z_k) \\
&= F_{z_k}(\emptyset) + \prod_{j=1}^{\nu'}(1 - q_j') \sum_{n=1}^{\nu'} \frac{1}{n!} \sum_{1 \le i_1 \ne \dots \ne i_n \le \nu'} \\
&\qquad \left( f_w * \int_{x_1} h(x_1)\, \xi_{i_1}\, \mathrm{d}x_1 * \dots * \int_{x_n} h(x_n)\, \xi_{i_n}\, \mathrm{d}x_n \right)(z_k) \\
&= F_{z_k}(\emptyset) + \prod_{j=1}^{\nu'}(1 - q_j') \sum_{n=1}^{\nu'} \frac{1}{n!} \sum_{1 \le i_1 \ne \dots \ne i_n \le \nu'} \left( f_w * \xi_{i_1}[h] * \dots * \xi_{i_n}[h] \right)(z_k)
\end{aligned}$$

with

$$\begin{aligned}
\xi_i[h](z) &= \int h(x_j)\, \xi_i\left( z \mid x_j \right) \mathrm{d}x_j \\
&= \int h(x_j)\, \frac{q_i'\, \Psi_j(z \mid x_j)\, s_i'(x_j)}{1 - q_i'}\, \mathrm{d}x_j \\
&= \frac{q_i' \int h(x_j)\, \Psi_j(z \mid x_j)\, s_i'(x_j)\, \mathrm{d}x_j}{1 - q_i'} \\
&= \frac{q_i'\, s_i'[h\, \Psi_j](z)}{1 - q_i'}
\end{aligned} \tag{7.2}$$

where $s_i'[h\, \Psi_j] \triangleq \int h(x_j)\, \Psi_j(z \mid x_j)\, s_i'(x_j)\, \mathrm{d}x_j$ and from (4.12) it is known that $\Psi_j(z \mid x_j) = \left(1 - p_V(x_j)\right)\delta_0(z) + p_V(x_j)\, \delta_{\eta(x_j)}(z)$. Since the convolution is commutative, the sum

$$\sum_{1 \le i_1 \ne \dots \ne i_n \le \nu'}$$

simplifies to

$$n! \sum_{1 \le i_1 < \dots < i_n \le \nu'}$$

and therefore $F_{z_k}[h]$ becomes

$$F_{z_k}[h] = F_{z_k}(\emptyset) + \prod_{j=1}^{\nu'}(1-q_j') \left( f_w * \sum_{n=1}^{\nu'} \sum_{1\le i_1<\ldots<i_n\le\nu'} \xi_{i_1}[h] * \ldots * \xi_{i_n}[h] \right)(z_k).$$

Recalling from (4.14) that

$$\sigma^*_{n,k}(f_1(z),\ldots,f_n(z)) = \begin{cases} \delta_0(z) & \text{if } k=0 \\ \sum\limits_{1\le i_1<\ldots<i_k\le n} \left(f_{i_1} * \ldots * f_{i_k}\right)(z) & \text{if } k>0 \\ 0 & \text{if } k>n \end{cases}$$

is the convoluted elementary symmetric function (CESF) which satisfies the identity

$$\sum_{k=0}^{n} \sigma^*_{n,k}(f_1(z),\ldots,f_n(z)) = \Big( (\delta_0+f_1) * \ldots * (\delta_0+f_n) \Big)(z),$$

then $F_{z_k}[h]$ simplifies to

$$F_{z_k}[h] = F_{z_k}(\emptyset) + \prod_{j=1}^{\nu'}(1-q_j') \left( f_w * \sum_{n=1}^{\nu'} \sigma^*_{\nu',n}(\xi_1[h],\ldots,\xi_{\nu'}[h]) \right)(z_k)$$

with

$$\begin{aligned} F_{z_k}(\emptyset) &= \prod_{j=1}^{\nu'}(1-q_j')\, f_w(z_k) \\ &\equiv \prod_{j=1}^{\nu'}(1-q_j')\, (f_w * \delta_0)(z_k) \end{aligned}$$

Consequently, this results in

$$\begin{aligned} F_{z_k}[h] &= \prod_{j=1}^{\nu'}(1-q_j') \left( (f_w*\delta_0)(z) + f_w * \sum_{n=1}^{\nu'} \sigma^*_{\nu',n}(\xi_1[h],\ldots,\xi_{\nu'}[h]) \right)(z_k) \\ &= \prod_{j=1}^{\nu'}(1-q_j') \left( f_w * \sum_{n=0}^{\nu'} \sigma^*_{\nu',n}(\xi_1[h],\ldots,\xi_{\nu'}[h]) \right)(z_k) \\ &= \prod_{j=1}^{\nu'}(1-q_j')\, (f_w * (\delta_0+\xi_1[h]) * \ldots * (\delta_0+\xi_{\nu'}[h]))(z_k). \end{aligned}$$

Finally, using (7.2) this becomes

$$F_{z_k}[h] = \prod_{j=1}^{\nu'}(1-q_j')\left(f_w * \left(\delta_0 + \frac{q_1'\, s_1'[h\,\Psi](z)}{1-q_1'}\right) * \ldots * \left(\delta_0 + \frac{q_{\nu'}'\, s_{\nu'}'[h\,\Psi](z)}{1-q_{\nu'}'}\right)\right)(z_k)$$
$$= \left(f_w * \left((1-q_1')\,\delta_0 + q_1'\, s_1'[h\,\Psi]\right) * \ldots * \left(\left(1-q_{\nu'}'\right)\delta_0 + q_{\nu'}'\, s_{\nu'}'[h\,\Psi]\right)\right)(z_k).$$

As a result, the nominator and denominator become

$$F_{z_k}[h] = (f_w * \zeta_1[h] * \ldots * \zeta_{\nu'}[h])(z_k)$$

and

$$F_{z_k}[1] = (f_w * \zeta_1[1] * \ldots * \zeta_{\nu'}[1])(z_k)$$

where

$$\zeta_i[h](z) = (1-q_i')\delta_0(z) + q_i'\, s_i'[h\,\Psi].$$

Furthermore, recalling (4.12) it follows that

$$\begin{aligned}
s_i'[h\,\Psi] &= \int h(x)\,\Psi(z \mid x)\; s_i'(x)\,\mathrm{d}x \\
&= \int h(x)\,\left((1-p_V(x))\,\delta_0(z) + \; p_V(x)\delta_{\eta(x)}(z)\right)\, s_i'(x)\,\mathrm{d}x \\
&= \int h(x)\;(1-p_V(x))\,\delta_0(z)\, s_i'(x)\,\mathrm{d}x + \int h(x)\, p_V(x)\delta_{\eta(x)}(z)\, s_i'(x)\,\mathrm{d}x \\
&= \int h(x)\;(1-p_V(x))\;\, s_i'(x)\,\mathrm{d}x\;\delta_0(z) + \int h(x)\, p_V(x)\delta_{\eta(x)}(z)\, s_i'(x)\,\mathrm{d}x \\
&= s_i'[h\,(1-p_V)]\;\delta_0(z) + s_i'\left[h\, p_V\;\delta_\eta\right](z),
\end{aligned}$$

this can be simplified to

$$\begin{aligned}
\zeta_i[h](z) &= (1-q_i')\;\delta_0(z) + q_i'\;\left(s_i'[h\,(1-p_V)]\;\delta_0(z) + s_i'\left[h\, p_V\;\delta_\eta\right](z)\right) \\
&= \left(1-q_i' + q_i'\, s_i'[h\,(1-p_V)]\right)\delta_0(z) + q_i'\, s_i'\left[h\, p_V\;\delta_\eta\right](z)
\end{aligned}$$

which after substituting $h(x) = 1$ becomes

$$\begin{aligned}
\zeta_i[1](z) &= \left(1-q_i' + q_i'\, s_i'[1-p_V]\right)\delta_0(z) + q_i'\, s_i'\left[p_V\;\delta_\eta\right](z) \\
&= \left(1-q_i' + q_i'\, s_i'[1] - q_i'\, s_i'[p_V]\right)\delta_0(z) + q_i'\, s_i'\left[p_V\;\delta_\eta\right](z).
\end{aligned}$$

Realizing that

$$s'[1] = \int 1\; s'(x)\,\mathrm{d}x = \int s'(x)\,\mathrm{d}x = 1$$

it reduces to

$$\zeta_i[1](z) = \left(1 - q_i' \, s_i'[p_V]\right)\delta_0(z) + q_i' \, s_i'\left[p_V \delta_\eta\right](z).$$

Finally, as claimed the PGFL of the multi–Bernoulli corrector for SPS–type sensors is

$$G_{k|k}[h \mid Z_{1:k}] = \frac{(f_w * \zeta_1[h] * \ldots * \zeta_{\nu'}[h])(z_k)}{(f_w * \zeta_1[1] * \ldots * \zeta_{\nu'}[1])(z_k)}$$

which concludes the proof of Theorem 1. □

The PDF form of the multi–Bernoulli corrector as stated in Theorem 2 can easily be derived from its PGFL form.

*Proof of Theorem 2.* Given the PGFL form of the SPS–type Bayes corrector, the PDF form becomes

$$\begin{aligned} f_{k|k}(X \mid Z_{1:k}) &= \frac{\delta G_{k|k}}{\delta X}[0 \mid Z_{1:k}] \\ &= \frac{1}{F_{z_k}[1]} \frac{\delta F_{z_k}}{\delta X}[0]. \end{aligned}$$

From (2.3) it is know that

$$\frac{\delta F_{z_k}}{\delta X}[0] = F_{z_k}(X)$$

and as such it follows

$$f_{k|k}(X \mid Z_{1:k}) = \frac{F_{z_k}(X)}{F_{z_k}[1]}.$$

From (7.1) $F_{z_k}(X)$ is already known and thus the PDF form of the Bayes corrector becomes

$$f_{k|k}(X \mid Z_{1:k}) = \frac{\prod_{j=1}^{\nu'}(1 - q_i') \sum_{1 \le i_1 \ne \ldots \ne i_n \le \nu'} \left(f_w * \xi_{i_1} * \ldots * \xi_{i_n}\right)(z_k)}{(f_w * \zeta_1[1] * \ldots * \zeta_{\nu'}[1])(z_k)}.$$

Furthermore, removing all constant terms from the equation it follows that

$$f_{k|k}(X \mid Z_{1:k}) \propto \sum_{1 \le i_1 \ne \ldots \ne i_n \le \nu'} \left(f_w * \xi_{i_1} * \ldots * \xi_{i_n}\right)(z_k).$$

Knowing that the PDF must integrate to 1 it follows

$$\int f_{k|k}(X \mid Z_{1:k}) \; \delta X = 1,$$

the PDF can be reformulated to

$$f_{k|k}(X \mid Z_{1:k}) = \frac{\sum\limits_{1 \leq i_1 \neq \ldots \neq i_n \leq \nu'} \left( f_w * \xi_{i_1} * \ldots * \xi_{i_n} \right)(z_k)}{\int \sum\limits_{1 \leq i_1 \neq \ldots \neq i_n \leq \nu'} \left( f_w * \xi_{i_1} * \ldots * \xi_{i_n} \right)(z_k) \, \delta X}.$$

Further, from (2.1) it follows that

$$\begin{aligned}
&\int \sum_{1 \leq i_1 \neq \ldots \neq i_n \leq \nu'} \left( f_w * \xi_{i_1} * \ldots * \xi_{i_n} \right)(z_k) \, \delta X \\
&\qquad = \sum_{n=0}^{\nu'} \frac{1}{n!} \int \cdots \int \sum_{1 \leq i_1 \neq \ldots \neq i_n \leq \nu'} \left( f_w * \xi_{i_1} * \ldots * \xi_{i_n} \right)(z_k) \, \mathrm{d}x_1 \cdots \mathrm{d}x_n \\
&\qquad = \sum_{n=0}^{\nu'} \frac{1}{n!} \sum_{1 \leq i_1 \neq \ldots \neq i_n \leq \nu'} \int \cdots \int \left( f_w * \xi_{i_1} * \ldots * \xi_{i_n} \right)(z_k) \, \mathrm{d}x_1 \cdots \mathrm{d}x_n.
\end{aligned}$$

Finally, when changing the order of integration and with

$$\begin{aligned}
\xi_i[1](z) &= \int \xi_i\left(z \mid x_j\right) \mathrm{d}x_j \\
&= \frac{q_i' \, s_i'[\Psi_j](z)}{1 - q_i'}
\end{aligned}$$

the denominator simplifies to

$$\sum_{n=0}^{\nu'} \frac{1}{n!} \sum_{1 \leq i_1 \neq \ldots \neq i_n \leq \nu'} \left( f_w * \xi_{i_1}[1] * \ldots * \xi_{i_n}[1] \right)(z_k)$$

Hence, the PDF results in

$$f_{k|k}(X \mid Z_{1:k}) = \frac{\sum\limits_{1 \leq i_1 \neq \ldots \neq i_n \leq \nu'} \left( f_w * \xi_{i_1} * \ldots * \xi_{i_n} \right)(z_k)}{\sum\limits_{n=0}^{\nu'} \frac{1}{n!} \sum\limits_{1 \leq i_1 \neq \ldots \neq i_n \leq \nu'} \left( f_w * \xi_{i_1}[1] * \ldots * \xi_{i_n}[1] \right)(z_k)}$$

and thus concludes the proof. □

### 7.2.2. Approximate $\Sigma$–MeMBer Filter

It has been shown in Section 7.2.1 that it is possible to derive the exact PGFL– and PDF–form of the Bayes corrector for SPS–type sensors. Unfortunately, it became evident that the a posteriori PDF is no longer the PDF of a multi–Bernoulli RFS. In the following sections the provided equations will be reformulated and the necessary steps will be taken to derive an approximate multi–Bernoulli PDF for the case of SPS–type sensors.

#### 7.2.2.1. Approximate $\Sigma$–MeMBer Filter Corrector

In this section, the approximate Bayes filter corrector equation for SPS–type sensor are presented.

**Theorem 3.** Let $\left\{\left(q_i', s_i'(x)\right)\right\}_{i=1}^{\nu'}$ be the predicted Bernoulli parameters, $z$ be the single SPS–type measurement, $f_w(z)$ be the likelihood that the measurement is generated by noise and $p_V(x)$ the probability that an object is detected/visible. Then, the corrected distribution is approximately multi–Bernoulli with parameters

$$\{(q_i, s_i(x))\}_{i=1}^{\nu} = \left\{\left(q_{m_i}, s_{m_i}(x)\right)\right\}_{i=1}^{\nu'} \cup \left\{\left(q_{d_i}(z), s_{d_i}(x \mid z)\right)\right\}_{i=1}^{\nu'}$$

where $q \overset{\text{abbr}}{=} q_{k|k}$, $s(x) \overset{\text{abbr}}{=} s_{k|k}(x)$ are the $\nu \overset{\text{abbr}}{=} \nu_{k|k}$ corrected components comprised out of the union of the components induced by missed objects $\left\{\left(q_{m_i}, s_{m_i}(x)\right)\right\}_{i=1}^{\nu'}$ determined by

$$q_{m_i} = \frac{1 - s_i'[p_V]}{1 - q_i'\, s_i'[p_V]}\, q_i'$$

and

$$s_{m_i}(x) = \frac{(1 - p_V(x))}{1 - s_i'[p_V]}\, s_i'(x),$$

and the Bernoulli parameters $\left\{\left(q_{d_i}(z), s_{d_i}(x \mid z)\right)\right\}_{i=1}^{\nu'}$ of the detected objects are

$$\begin{aligned} q_{d_i}(z) &= \frac{(1 - q_i')}{\left(1 - q_i'\, s_i'[p_V]\right)}\, \frac{q_i'\, s_i'\left[p_V\ \varphi_\eta^{\neq i}\right](z)}{\left(1 - q_i'\, s_i'[p_V]\right)\ \varphi^{\neq i}(z) + q_i'\, s_i'\left[p_V\ \varphi_\eta^{\neq i}\right](z)} \\ &\equiv \frac{\frac{(1' - q_i)}{\left(1 - q_i'\, s_i'[p_V]\right)}\ \frac{q_i'\, s_i'\left[p_V\ \varphi_\eta^{\neq i}\right](z)}{\left(1 - q_i'\, s_i'[p_V]\right)}}{\varphi^{\neq i}(z) + \frac{q_i'\, s_i'\left[p_V\ \varphi_\eta^{\neq i}\right](z)}{1 - q_i'\, s_i'[p_V]}} \end{aligned}$$

and

$$s_{d_i}(x \mid z) = \frac{p_V(x)\ \varphi^{\neq i}(z - \eta(x))\ s_i'(x)}{s_i'\left[p_V\ \varphi_\eta^{\neq i}\right]}$$

with $s_i'[h] \triangleq \int h(x)\, s_i'(x)\, \mathrm{d}x$ and $\varphi_\eta^{\neq i}(z \mid x) \triangleq \varphi^{\neq i}(z - \eta(x))$. The Pseudo–likelihood is

$$\varphi^{\neq i}(z) \triangleq (f_w * \zeta_1[1] * \ldots * \zeta_{i-1}[1] * \zeta_{i+1}[1] * \ldots * \zeta_{\nu'}[1])(z)$$

with

$$\zeta_i[1](z) \triangleq \left(1 - q_i'\, s_i'[p_V]\right)\delta_0(z) + q_i' s_i'\left[p_V\ \delta_\eta\right](z).$$

*Proof.* See Section 7.2.2.2.

While the approximate SPS–type MeMBer corrector equations presented in Theorem 3 look rather efficient at first glance, the computationally demanding part is the determination of the Pseudo–likelihood $\varphi^{\neq i}(z-\eta(x))$ since it involves multiple convolutions of $\zeta_i(z)$. In fact the Pseudo–likelihood $\varphi^{\neq j}$ is a mixture of $2^{\nu'-1}$ components and has to be generated $\nu'$ times. While these equations can be used they are still too computationally demanding when the number of Bernoulli components and such the number of possible objects gets larger. After providing an alternative MeMBer corrector for SPS sensors in Section 7.2.3, several options to reduce the computational demand in determining the Pseudo–likelihood $\varphi^{\neq i}(z-\eta(x))$ will be presented in Chapter 8.

#### 7.2.2.2. Derivation of the Approximate $\Sigma$–MeMBer Filter Corrector

In this section, the principal equations for the approximate MeMBer corrector equations as presented in Theorem 3 are proven. In order to proof the results stated in Theorem 3, the PGFL variant of the SPS–type MeMBer filter corrector as stated in Theorem 1 is factorized first. Recall the PGFL variant of the SPS–type MeMBer filter corrector as stated in Theorem 1

$$G[h \mid Z_{1:k}] = \frac{(f_w * \zeta_1[h] * \ldots * \zeta_{\nu'}[h])(z)}{(f_w * \zeta_1[1] * \ldots * \zeta_{\nu'}[1])(z)}$$

with

$$\zeta_i[h](z) \triangleq \left(1 - q_i' + q_i' s_i'[h\,(1-p_V)]\right)\delta_0(z) + q_i' s_i'\left[h\; p_V\; \delta_\eta\right](z)$$

and

$$\zeta_i[1](z) \triangleq \left(1 - q_i' s_i'[p_V]\right)\delta_0(z) + q_i' s_i'\left[p_V\; \delta_\eta\right](z).$$

Let $\zeta_i[h](z)$ be factorized as

$$\zeta_i[h](z) = \zeta_{m_i}[h]\, \zeta_{d_i}[h](z)$$

and

$$\zeta_i[1](z) = \zeta_{m_i}[1]\, \zeta_{d_i}[1](z)$$

where the terms

$$\zeta_{m_i}[h] = \left(1 - q_i' + q_i' s_i'[h\,(1-p_V)]\right)$$

and

$$\zeta_{m_i}[1] = \left(1 - q_i' s_i'[p_V]\right)$$

correspond to the $i$–th missed objects. Furthermore, it follows that

$$\begin{aligned}\zeta_{d_i}[h](z) &= \frac{\zeta_i[h](z)}{\zeta_{m_i}[h]} \\ &= \frac{\left(1 - q_i' + q_i'\, s_i'[h\,(1-p_V)]\right)\delta_0(z) + q_i' s_i'\left[h\; p_V\; \delta_\eta\right](z)}{1 - q_i' + q_i'\, s_i'[h\,(1-p_V)]} \\ &= \left(\delta_0(z) + \frac{q_i' s_i'\left[h\; p_V\; \delta_\eta\right](z)}{1 - q_i' + q_i'\, s_i'[h\,(1-p_V)]}\right)\end{aligned}$$

and

$$\begin{aligned}\zeta_{d_i}[1](z) &= \frac{\zeta_i[1](z)}{\zeta_{m_i}[1]} \\ &= \frac{\left(1 - q_i'\, s_i'[p_V]\right)\delta_0(z) + q_i' s_i'\left[p_V\; \delta_\eta\right](z)}{1 - q_i'\, s_i'[p_V]} \\ &= \left(\delta_0(z) + \frac{q_i' s_i'\left[p_V\; \delta_\eta\right](z)}{1 - q_i'\, s_i'[p_V]}\right)\end{aligned}$$

correspond to the $i$–th detected object. Since the missed object part $\zeta_{m_i}[h]$ is no function of $z$ it can be moved out of the convolution and the corrector becomes

$$G[h \mid Z_{1:k}] = \frac{\prod_{i=1}^{\nu'} \zeta_{m_i}[h]\, \left(f_w * \zeta_{d_1}[h] * \ldots * \zeta_{d_{\nu'}}[h]\right)(z)}{\prod_{i=1}^{\nu'} \zeta_{m_i}[1]\, \left(f_w * \zeta_{d_1}[1] * \ldots * \zeta_{d_{\nu'}}[1]\right)(z)}.$$

Consequently, when defining

$$G_m[h \mid Z_{1:k}] = \prod_{i=1}^{\nu'} \frac{1 - q_i' + q_i'\, s_i'[h\,(1-p_V)]}{1 - q_i'\, s_i'[p_V]} \tag{7.3}$$

to be the PGFL of the new components induced by missed objects and

$$G_d[h \mid Z_{1:k}] = \frac{\left(f_w * \zeta_{d_1}[h] * \ldots * \zeta_{d_{\nu'}}[h]\right)(z)}{\left(f_w * \zeta_{d_1}[1] * \ldots * \zeta_{d_{\nu'}}[1]\right)(z)} \tag{7.4}$$

to be the PGFL of the detected or visible objects, it follows that the PGFL can be factorized as

$$G[h \mid Z_{1:k}] = G_m[h \mid Z_{1:k}]\; G_d[h \mid Z_{1:k}].$$

Note that if it can be shown that $G_m[h \mid Z_{1:k}]$ and $G_d[h \mid Z_{1:k}]$ are both PGFLs of multi–Bernoulli RFSs, then $G[h \mid Z_{1:k}]$ is consequently also a PGFL of a multi–Bernoulli RFS.

*Proof of Theorem 3.* In the following it will be proven that $G[h \mid Z_{1:k}]$ is the PGFL of a multi–Bernoulli RFS. In order to prove this, it will be shown that (7.3) and (7.4) are both valid PGFLs of multi–Bernoulli RFS. It will be made use of the fact that the product of the PGFLs of two Bernoulli RFSs is in fact a PGFL of a multi–Bernoulli RFS. Furthermore, the product of the PGFLs of two multi–Bernoulli RFSs is also a multi–Bernoulli RFS (see. (3.1)). Recalling (7.3) and defining

$$G_{m_i}[h \mid Z_{1:k}] = \frac{1 - q_i' + q_i' \, s_i'[h\,(1 - p_V)]}{1 - q_i' \, s_i'[p_V]}$$

results in

$$G_m[h \mid Z_{1:k}] = \prod_{i=1}^{\nu'} G_{m_i}[h \mid Z_{1:k}].$$

Thus, in order to proof that $G_m[h \mid Z_{1:k}]$ is a valid PGFL of a multi–Bernoulli RFS, it only needs to be shown that $G_{m_i}[h \mid Z_{1:k}]$ is a valid PGFL of a Bernoulli RFS. Similar to [Mah07b, Sec. 17.4.2.3], it follows that

$$\begin{aligned} G_{m_i}[h \mid Z_{1:k}] &= \frac{1 - q_i' + q_i' \, s_i'[h\,(1 - p_V)]}{1 - q_i' \, s_i'[p_V]} \\ &= \frac{1 - q_i'}{1 - q_i' \, s_i'[p_V]} + \frac{q_i' \, s_i'[h\,(1 - p_V)]}{1 - q_i' \, s_i'[p_V]}. \end{aligned}$$

In turn, when extending the term with $\frac{s_i'[(1-p_V)]}{s_i'[(1-p_V)]}$, this becomes

$$G_{m_i}[h \mid Z_{1:k}] = \frac{1 - q_i'}{1 - q_i' \, s_i'[p_V]} + \frac{q_i' \, s_i'[(1 - p_V)]}{1 - q_i' \, s_i'[p_V]} \frac{s_i'[h\,(1 - p_V)]}{s_i'[(1 - p_V)]}.$$

Then, adding $\pm q_i' \, s_i'[p_V]$ results in

$$\begin{aligned} G_{m_i}[h \mid Z_{1:k}] &= \frac{1 - q_i' - q_i' \, s_i'[p_V] + q_i' \, s_i'[p_V]}{1 - q_i' \, s_i'[p_V]} + \frac{q_i' \, s_i'[(1 - p_V)]}{1 - q_i' \, s_i'[p_V]} \frac{s_i'[h\,(1 - p_V)]}{s_i'[(1 - p_V)]} \\ &= \frac{1 - q_i' \, s_i'[p_V]}{1 - q_i' \, s_i'[p_V]} - \frac{q_i' - q_i' \, s_i'[p_V]}{1 - q_i' \, s_i'[p_V]} + \frac{q_i' \, s_i'[(1 - p_V)]}{1 - q_i' \, s_i'[p_V]} \frac{s_i'[h\,(1 - p_V)]}{s_i'[(1 - p_V)]} \\ &= 1 - \frac{q_i' \, s_i'[(1 - p_V)]}{1 - q_i' \, s_i'[p_V]} + \frac{q_i' \, s_i'[(1 - p_V)]}{1 - q_i' \, s_i'[p_V]} \, \frac{s_i'[h\,(1 - p_V)]}{s_i'[(1 - p_V)]}. \end{aligned}$$

Finally, this leads to the typical PGFL form of a Bernoulli RFS when defining

$$G_{m_i}[h \mid Z_{1:k}] = 1 - q_{m_i} + q_{m_i} \, s_{m_i}[h]$$

with

$$\begin{aligned} q_{m_i} &\triangleq \frac{q_i' \, s_i'[(1 - p_V)]}{1 - q_i' \, s_i'[p_V]} \\ &= \frac{q_i' - q_i' \, s_i'[p_V]}{1 - q_i' \, s_i'[p_V]} \end{aligned}$$

and

$$s_{m_i}(x) = \frac{(1 - p_V)\, s_i'(x)}{s_i'[(1 - p_V)]},$$

where $s_i'[h] = \int h(x)\, s_i'(x)\, \mathrm{d}x$. Since $G_{m_i}[h \mid Z_{1:k}]$ is the PGFL of a Bernoulli RFS it naturally follows that $G_m[h \mid Z_{1:k}]$ is the PGFL of a multi–Bernoulli distribution as described in Section 3.4.

Additionally, it needs to be proven that $G_d[h \mid Z_{1:k}]$ is the PGFL of a multi–Bernoulli RFS. While it is not possible to show that $G_d[h \mid Z_{1:k}]$ is the PGFL of a multi–Bernoulli RFS, it is indeed possible to derive an approximation $\tilde{G}_d[h \mid Z_{1:k}]$ that is the PGFL of a multi–Bernoulli RFS. Here, the approximation $\tilde{G}_d[h \mid Z_{1:k}]$ is derived such that the PHD of the new distribution and the original are equal. Hence, the resulting approximation is accurate up to the first factorial moment. In order to achieve this, the corresponding PHD $D_d(x \mid Z_{1:k})$ of $G_d[h \mid Z_{1:k}]$ will be determined, which in turn will make it possible to derive approximate Bernoulli parameters for the detected objects. By definition the PHD follows to be

$$D_d(x \mid Z_{1:k}) = \frac{\delta G_d}{\delta x}[1 \mid Z_{1:k}].$$

Consequently, when

$$\begin{aligned} K_d &\triangleq \left(f_w * \zeta_{d_1}[1] * \ldots * \zeta_{d_{\nu'}}[1]\right)(z) \\ &= \left(\varphi^{\neq d_j} * \zeta_{d_j}[1]\right)(z) \end{aligned} \tag{7.5}$$

and due to the product rule of functional derivatives (app. A) this becomes

$$D_d(x \mid Z_{1:k}) = {K_d}^{-1} \sum_{j=1}^{\nu'} \left(f_w * \zeta_{d_1}[1] * \ldots * \frac{\delta \zeta_{d_j}}{\delta x}[1] * \ldots * \zeta_{d_{\nu'}}[1]\right)(z).$$

Further, let

$$\begin{aligned} \varphi^{\neq d_j}(z) &\triangleq \left(f_w * \zeta_{d_1}[1] * \ldots * \zeta_{d_{j-1}}[1] * \zeta_{d_{j+1}}[1] * \ldots * \zeta_{d_{\nu'}}[1]\right)(z) \\ &= \left(f_w * \frac{\zeta_1[1]}{\zeta_{m_1}[1]} * \ldots * \frac{\zeta_{j-1}[1]}{\zeta_{m_{j-1}}[1]} * \frac{\zeta_{j+1}[1]}{\zeta_{m_{j+1}}[1]} * \ldots * \frac{\zeta_{\nu'}[1]}{\zeta_{m_{\nu'}}[1]}\right)(z) \\ &= \frac{\zeta_{m_j}[1]}{\prod_{i=1}^{\nu'} \zeta_{m_i}[1]} \left(f_w * \zeta_1[1] * \ldots * \zeta_{j-1}[1] * \zeta_{j+1}[1] * \ldots * \zeta_{\nu'}[1]\right)(z) \end{aligned}$$

to be the convolution of all $\zeta_{d_i}[1](z)$ but not the $j$–th element and recalling that

$$\varphi^{\neq j}(z) = \left(f_w * \zeta_1[1] * \ldots * \zeta_{j-1}[1] * \zeta_{j+1}[1] * \ldots * \zeta_{\nu'}[1]\right)(z)$$

is the convolution of all $\zeta_i[1](z)$ with $i \in \{1,\dots,\nu'\} \setminus \{j\}$, then it follows that

$$\varphi^{\neq d_j}(z) = \frac{\zeta_{m_j}[1]}{\prod_{i=1}^{\nu'} \zeta_{m_i}[1]}\, \varphi^{\neq j}(z). \tag{7.6}$$

Consequently, it follows that

$$\begin{aligned} D_d(x \mid Z_{1:k}) &= \sum_{j=1}^{\nu'} {K_d}^{-1} \left(\varphi^{\neq d_j} * \frac{\delta \zeta_{d_j}}{\delta x}[1]\right)(z) \\ &= \sum_{j=1}^{\nu'} D_{d_j}(x \mid Z_{1:k}) \end{aligned} \tag{7.7}$$

where

$$D_{d_j}(x \mid Z_{1:k}) = {K_d}^{-1} \left(\varphi^{\neq d_j} * \frac{\delta \zeta_{d_j}}{\delta x}[1]\right)(z). \tag{7.8}$$

After applying the product rule for functional derivatives, the functional derivative of $\zeta_{d_j}[h]$ follows to be

$$\begin{aligned} \frac{\delta \zeta_{d_j}[h](z)}{\delta x} &= \frac{\delta\left(\zeta_j[h](z) / \zeta_{m_j}[h]\right)}{\delta x} \\ &= \zeta_{m_j}[h]^{-2} \left(\frac{\delta \zeta_j[h](z)}{\delta x}\, \zeta_{m_j}[h] - \zeta_j[h](z)\, \frac{\delta \zeta_{m_j}[h]}{\delta x}\right) \end{aligned} \tag{7.9}$$

where the functional derivatives of $\zeta_j[h](z)$ and $\zeta_{m_j}[h]$ are

$$\begin{aligned} \frac{\delta \zeta_j[h](z)}{\delta x} &= \frac{\delta\left(\left(1 - q_j' + q_j'\, s_j'[h\,(1-p_V)]\right)\delta_0(z)\right)}{\delta x} + \frac{\delta\left(q_j'\, s_j'\left[h\, p_V\, \delta_\eta\right](z)\right)}{\delta x} \\ &= \left(q_j'\, \frac{\delta\left(s_j'[h\,(1-p_V)]\right)}{\delta x}\right)\delta_0(z) + q_j'\, \frac{\delta\left(s_j'\left[h\, p_V\, \delta_\eta\right](z)\right)}{\delta x} \\ &= q_j'\,(1-p_V(x))\, s_j'(x)\, \delta_0(z) + q_j'\, p_V(x)\, \delta_{\eta(x)}(z)\, s_j'(x) \end{aligned}$$

and

$$\begin{aligned} \frac{\delta \zeta_{m_j}[h]}{\delta x} &= \frac{\delta\left(1 - q_j' + q_j'\, s_j'[h\,(1-p_V)]\right)}{\delta x} \\ &= q_j'\, \frac{\delta\left(s_j'[h\,(1-p_V)]\right)}{\delta x} \\ &= q_j'\,(1-p_V(x))\, s_j'(x). \end{aligned}$$

Finally, after reinserting these into (7.9) the result is

$$\begin{aligned}\frac{\delta\zeta_{d_j}[h]}{\delta x} = &\left(1 - q_j' + q_j'\, s_j'[h\,(1-p_V)]\right)^{-2} \\ &\Big(q_j'\, p_V(x)\, \delta_{\eta(x)}(z)\, s_j'(x)\, \left(1 - q_j' + q_j'\, s_j'[h\,(1-p_V)]\right) \\ &\quad - q_j'\, s_j'\Big[h\; p_V\; \delta_\eta\Big](z)\; q_j'\,(1-p_V(x))\, s_j'(x)\Big),\end{aligned}$$

which when setting $h = 1$ simplifies to

$$\begin{aligned}\frac{\delta\zeta_{d_j}}{\delta x}[1] = &\left(1 - q_j'\, s_j'[p_V]\right)^{-2} \\ &\Big(q_j'\, p_V(x)\delta_{\eta(x)}(z)\, s_j'(x)\, \left(1 - q_j'\, s_j'[p_V]\right) \\ &\quad - q_j'\, s_j'\Big[p_V\; \delta_\eta\Big](z)\; q_j'\,(1-p_V(x))\, s_j'(x)\Big).\end{aligned} \tag{7.10}$$

Consequently, after inserting (7.10) into (7.8) the PHD $D_{d_j}(x \mid Z_{1:k})$ of the PGFL $G_d[h \mid Z_{1:k}]$ becomes

$$\begin{aligned}D_{d_j}(x \mid Z_{1:k}) = {K_d}^{-1} &\Big(\varphi^{\neq d_j} * \left(1 - q_j'\, s_j'[p_V]\right)^{-2} \\ &\Big(q_j'\, p_V(x)\, \delta_{\eta(x)}(z)\, s_j'(x)\, \left(1 - q_j'\, s_j'[p_V]\right) \\ &\quad - q_j'\, s_j'\Big[p_V\; \delta_\eta\Big](z)\; q_j'\,(1-p_V(x))\, s_j'(x)\Big).\end{aligned}$$

Note that $\varphi^{\neq d_j}(z)$ and $\delta_{\eta(x)}(z)$ are the only terms that are dependent on the measurement $z$ and $s_j'\Big[p_V\; \delta_\eta\Big](z) = \int p_V(x)\, \delta_\eta(z)\, s_j'(x)\mathrm{d}x$. Hence, with

$$\begin{aligned}\varphi^{\neq d_j}(z) * s_j'\Big[p_V\; \delta_\eta\Big](z) &= \varphi^{\neq d_j}(z) * \int p_V(x)\, \delta_{\eta(x)}(z)\, s_j'(x)\, \mathrm{d}x \\ &= \int \varphi^{\neq d_j}(\tau)\left(\int p_V(x)\, \delta_{\eta(x)}(z-\tau)\, s_j'(x)\, \mathrm{d}x\right)\mathrm{d}\tau \\ &= \int p_V(x)\left(\int \varphi^{\neq d_j}(\tau)\, \delta_{\eta(x)}(z-\tau)\mathrm{d}\tau\right) s_j'(x)\, \mathrm{d}x \\ &= \int p_V(x)\left(\varphi^{\neq d_j}(z) * \delta_\eta(z)\right) s_j'(x)\, \mathrm{d}x \\ &= s_j'\Big[p_V\, (\varphi^{\neq d_j} * \delta_\eta)\Big](z),\end{aligned}$$

this becomes

$$D_{d_j}(x \mid Z_{1:k}) = {K_d}^{-1} \left(1 - q'_j \, s'_j[p_V]\right)^{-2} \left(q'_j \, p_V(x) \, (\varphi^{\neq d_j} * \delta_\eta)(z) \, s'_j(x) \left(1 - q'_j \, s'_j[p_V]\right) - q'_j \, s'_j\Big[p_V \, (\varphi^{\neq d_j} * \delta_\eta)\Big](z) \; q'_j \, (1 - p_V(x)) \, s'_j(x)\right)$$

Finally, with $\varphi_\eta^{\neq d_j}(z \mid x) \triangleq (\varphi^{\neq d_j} * \delta_\eta)(z) = \varphi^{\neq d_j}(z - \eta(x))$ this results in

$$D_{d_j}(x \mid Z_{1:k}) = {K_d}^{-1} \left(1 - q'_j \, s'_j[p_V]\right)^{-2} \left(q'_j \, p_V(x) \, \varphi_\eta^{\neq d_j}(z \mid x) \, s'_j(x) \left(1 - q'_j \, s'_j[p_V]\right) - q'_j \, s'_j\Big[p_V \; \varphi_\eta^{\neq d_j}\Big](z) \; q'_j \, (1 - p_V(x)) \, s'_j(x)\right). \tag{7.11}$$

After deriving the PHD, this information can be used to determine a first order approximation of the Bernoulli parameters. Recalling the general structure of the PHD of a multi–Bernoulli RFS with $\nu'$ components being

$$D(x) = \sum_{j=1}^{\nu'} q_j \, s_j(x)$$

and comparing it to the previous result, then it follows

$$\begin{aligned} D_d(x) &= \sum_{j=1}^{\nu'} D_{d_j}(x \mid Z_{1:k}) \\ &= \sum_{j=1}^{\nu'} q_{d_j} \, s_{d_j}(x). \end{aligned}$$

Comparing the general structure of the PHD of a multi–Bernoulli RFS to (7.7) and noticing that both are the sum over the individual $\nu'$ Bernoulli components, it is chosen to approximate the Bernoulli parameters individually. Thus, it follows

$$D_{d_j}(x \mid Z_{1:k}) \approx q_{d_j}(z) \, s_{d_j}(x \mid z).$$

Since, $s_{d_j}(x \mid z)$ is a PDF it is known that $\int s_{d_j}(x \mid z) \, \mathrm{d}x = 1$. Thus, it is possible to derive an approximation for the probability of existence by integrating over the

PHD, such that

$$\begin{aligned}\int D_{d_j}(x \mid Z_{1:k})\,\mathrm{d}x &\approx \int q_{d_j}(z)\, s_{d_j}(x \mid z)\,\mathrm{d}x \\ &= q_{d_j}(z) \int s_{d_j}(x \mid z)\,\mathrm{d}x \\ &= q_{d_j}(z).\end{aligned}$$

Furthermore, the PDF can also be derived from the PHD by

$$s_{d_j}(x \mid z) \approx q_{d_j}^{-1}\, D_{d_j}(x \mid Z_{1:k}).$$

Subsequently, the approximate update equation for the probability of existence are proven. Based on the previous insights the approximate probability of existence $q_{d_j}(z)$ is

$$\begin{aligned}q_{d_j}(z) &\approx \int D_{d_j}(x \mid Z_{1:k})\,\mathrm{d}x \\ &= {K_d}^{-1} \left(1 - q_j' \, s_j'[p_V]\right)^{-2} \\ &\quad \left(q_j' \, s_j'\left[p_V \, \varphi_\eta^{\neq d_j}\right](z) \left(1 - q_j' \, s_j'[p_V]\right) - q_j' \, s_j'\left[p_V \, (\varphi_\eta^{\neq d_j})\right](z)\, q_j' \, s_j'[1 - p_V]\right) \\ &= {K_d}^{-1} \frac{q_j' \, s_j'\left[p_V \, \varphi_\eta^{\neq d_j}\right](z)}{\left(1 - q_j' \, s_j'[p_V]\right)^2} \left(1 - \cancel{q_j' \, s_j'[p_V]} - q_j' + \cancel{q_j' \, s_j'[p_V]}\right) \\ &= {K_d}^{-1} \frac{q_j' \left(1 - q_j'\right) s_j'\left[p_V \, \varphi_\eta^{\neq d_j}\right](z)}{\left(1 - q_j' \, s_j'[p_V]\right)^2}. \end{aligned} \tag{7.12}$$

When $K_d$ is replaced by its definition from (7.5), then $q_{d_j}$ becomes

$$\begin{aligned}q_{d_j}(z) &\approx \frac{\frac{q_j' \left(1 - q_j'\right) s_j'\left[p_V \, \varphi_\eta^{\neq d_j}\right](z)}{\left(1 - q_j' \, s_j'[p_V]\right)^2}}{\left(\varphi^{\neq d_j} * \zeta_{d_j}[1]\right)(z)} \\ &= \frac{\frac{q_j' \left(1 - q_j'\right) s_j'\left[p_V \, \varphi_\eta^{\neq d_j}\right](z)}{\left(1 - q_j' \, s_j'[p_V]\right)^2}}{\left(\varphi^{\neq d_j} * \left(\delta_0 + \frac{q_j' s_j'[p_V \, \delta_\eta]}{1 - q_j' \, s_j'[p_V]}\right)\right)(z)}.\end{aligned}$$

Since the convolution is distributive the denominator can be simplified. That is

why, it follows that

$$q_{d_j}(z) = \frac{\frac{q'_j\,(1-q'_j)\,s'_j\left[p_V\;\varphi_\eta^{\neq d_j}\right](z)}{\left(1-q'_j\,s'_j[p_V]\right)^2}}{\frac{\left(1-q'_j\,s'_j[p_V]\right)\varphi^{\neq d_j}(z)+q'_j s'_j\left[p_V\;\varphi_\eta^{\neq d_j}\right](z)}{1-q'_j\,s'_j[p_V]}}$$

$$= \frac{(1-q'_j)}{\left(1-q'_j\,s'_j[p_V]\right)}\;\frac{q'_j\,s'_j\left[p_V\;\varphi_\eta^{\neq d_j}\right](z)}{\left(1-q'_j\,s'_j[p_V]\right)\varphi^{\neq d_j}(z)+q'_j s'_j\left[p_V\;\varphi_\eta^{\neq d_j}\right](z)}.$$

Further, from (7.6) it is known that

$$\varphi^{\neq d_j}(z) = \frac{\zeta_{m_j}[1]}{\prod_{i=1}^{\nu'}\zeta_{m_i}[1]}\;\varphi^{\neq j}(z)$$

and as such the single–component probability of existence $q_{d_j}$ becomes as proposed

$$q_{d_j}(z) = \frac{(1-q'_j)}{\left(1-q'_j\,s'_j[p_V]\right)}\;\frac{q'_j\,s'_j\left[p_V\;\varphi_\eta^{\neq j}\right]}{\left(1-q'_j\,s'_j[p_V]\right)\varphi^{\neq j}(z)+q'_j s'_j\left[p_V\;\varphi_\eta^{\neq j}\right]}.$$

Conforming to the same principle the approximate PDF can be proven to be as follows. The approximate PDF $s_j(x\mid z)$ becomes

$$s_{d_j}(x\mid z) = \frac{D_{d_j}(x\mid Z_{1:k})}{q_{d_j}(z)}.$$

Hence, after inserting (7.11) this leads to

$$s_{d_j}(x\mid z) = \frac{K_D^{-1}}{q_{d_j}(z)}\;\left(1-q'_j\,s'_j[p_V]\right)^{-2}$$
$$\Big(q'_j\,\varphi^{\neq d_j}(z-\eta(x))\;p_V(x)\;s'_j(x)\;\left(1-q'_j\,s'_j[p_V]\right)$$
$$-\,q'_j\,s'_j\left[p_V\;\varphi_\eta^{\neq d_j}\right](z)\;q'_j\,s'_j(x)\;(1-p_V(x))\Big).$$

Further, realizing from (7.12) that

$$\frac{K_d^{-1}}{q_{d_j}(z)} = \frac{\left(1-q'_j\,s'_j[p_V]\right)^2}{q'_j\,(1-q'_j)\,s'_j\left[p_V\;\varphi_\eta^{\neq d_j}\right](z)}$$

leads to

$$
\begin{aligned}
s_{d_j}(x \mid z) \approx & \frac{\cancel{\left(1 - q_j' \, s_j'[p_V]\right)^2}}{q_j' \, (1 - q_j') \, s_j'\left[p_V \; \varphi_\eta^{\neq d_j}\right](z)} \cancel{\left(1 - q_j' \, s_j'[p_V]\right)^{-2}} \\
& \Big( q_j' \, \varphi^{\neq d_j}(z - \eta(x)) \, p_V(x) s_j'(x) \, \left(1 - q_j' \, s_j'[p_V]\right) \\
& \quad - q_j' \, s_j'\left[p_V \; \varphi_\eta^{\neq d_j}\right](z) \, q_j' \, (1 - p_V(x)) \, s_j'(x) \Big) \\
= & \frac{(1 - q_j' \, s[p_V]) \, q_j' \, p_V(x) \, \varphi^{\neq d_j}(z - \eta(x)) \, s_j'(x)}{q_j' \, (1 - q_j') \, s_j'\left[p_V \; \varphi_\eta^{\neq d_j}\right](z)} - \frac{q_j' \, (1 - p_V(x)) \, s_j'(x)}{1 - q_j'} \\
= & \left( \frac{(1 - q_j' \, s[p_V]) \, p_V(x) \, \varphi^{\neq d_j}(z - \eta(x))}{(1 - q_j') \, s_j'\left[p_V \; \varphi_\eta^{\neq d_j}\right](z)} - \frac{q_j' \, (1 - p_V(x))}{1 - q_j'} \right) s_j'(x) \\
= & \, L_{d_j}(z - \eta(x)) \, s_j'(x)
\end{aligned}
$$

where the Pseudo–likelihood $L_{d_j}(z - \eta(x))$ is defined as

$$
L_{d_j}(z - \eta(x)) \triangleq \frac{(1 - q_j' \, s[p_V]) \, p_V(x) \, \varphi^{\neq d_j}(z - \eta(x))}{(1 - q_j') \, s_j'\left[p_V \; \varphi_\eta^{\neq d_j}\right](z)} - \frac{q_j' \, (1 - p_V(x))}{1 - q_j'}.
$$

Nevertheless, the likelihood might become negative for $p_V(x) < 1$ since the term $-\frac{q_j' \, (1 - p_V(x))}{1 - q_j'}$ is always negative because $q_j' > 0$ and $p_V(x) \geq 0 \; \forall x \in \mathcal{X}$ and therefore the update would result in an invalid PDF $s_{d_j}(x)$.

Note that while the previous result is correct in terms of the math, the resulting Pseudo–likelihood is only a valid likelihood if $p_V(x) = 1$. Setting $p_V(x) = 1$ results in

$$
\begin{aligned}
L_{d_j}(z - \eta(x)) \overset{p_V(x)=1}{=} & \frac{(1 - q_j' \, s[1]) \, 1 \, \varphi^{\neq d_j}(z - \eta(x))}{(1 - q_j') \, s_j'\left[1 \; \varphi_\eta^{\neq d_j}\right](z)} - \frac{q_j' \, (1 - 1)}{1 - q_j'} \\
= & \frac{\cancel{(1 - q_j')} \, \varphi^{\neq d_j}(z - \eta(x))}{\cancel{(1 - q_j')} \, s_j'\left[\varphi_\eta^{\neq d_j}\right](z)} - \cancel{\frac{q_j' \, (1 - 1)}{1 - q_j'}} \\
= & \frac{\varphi^{\neq d_j}(z - \eta(x))}{s_j'\left[\varphi_\eta^{\neq d_j}\right](z)}.
\end{aligned}
$$

In order to derive a valid PDF $s_{d_j}(x)$ it is chosen to drop the term $\frac{q'_j\,(1-p_V(x))}{1-q'_j}$. Thus, the Pseudo–likelihood becomes

$$\begin{aligned} L_{d_j}(z-\eta(x)) &\leq \frac{(1-q'_j\,s[p_V])\,p_V(x)\,\varphi^{\neq d_j}(z-\eta(x))}{(1-q'_j)\,s'_j\left[p_V\;\varphi_\eta^{\neq d_j}\right](z)} \\ &\triangleq \tilde{L}_{d_j}(z-\eta(x)), \end{aligned}$$

which is an upper bound for all possible $x \in \mathcal{X}$. It follows that the integral

$$\int \tilde{L}_{d_j}(z-\eta(x))\,s'_j(x)\,\mathrm{d}x \geq 1$$

and as such renormalization is necessary. It follows that after renormalization the approximate PDF becomes

$$\begin{aligned} s_{d_j}(x \mid z) &\approx \frac{\tilde{L}_{d_j}(z-\eta(x))\,s'_j(x)}{\int \tilde{L}_{d_j}(z-\eta(x))\,s'_j(x)\,\mathrm{d}x} \\ &= \frac{p_V(x)\,\varphi^{\neq d_j}(z-\eta(x))\,s'_j(x)}{\int p_V(x)\,\varphi^{\neq d_j}(z-\eta(x))\,s'_j(x)\,\mathrm{d}x}. \end{aligned}$$

Realizing from (7.6) that $\varphi^{\neq d_j}(z) = \frac{\zeta_{m_j}[1]}{\prod_{i=1}^{\nu'}\zeta_{m_i}[1]}\,\varphi^{\neq j}(z)$, then it follows as claimed

$$s_{d_j}(x \mid z) \approx \frac{p_V(x)\,\varphi^{\neq j}(z-\eta(x))\,s'_j(x)}{\int p_V(x)\,\varphi^{\neq j}(z-\eta(x))\,s'_j(x)\,\mathrm{d}x},$$

which finally concludes the proof. □

### 7.2.3. Intensity $\Sigma$–MeMBer Filter

Besides generating a valid approximation to the exact SPS–type MeMBer corrector by factorizing out the missed detections and then finding an approximation of the detected Bernoulli components with the help of its PHD as described in Section 7.2.2, it is also possible to infer approximate Bernoulli parameters directly from the exact PGFL provided in Theorem 1. While this approximation is more crude than the other one, it is not obvious whether it will perform better or worse.

#### 7.2.3.1. Intensity $\Sigma$–MeMBer Corrector

In this section, the principal equations of the intensity SPS–type MeMBer filter corrector are provided.

**Theorem 4.** Given the set $\left\{\left(q_i', s_i'(x)\right)_{i=1}^{\nu'}\right\}$ of predicted multi–Bernoulli components, $q_i'$ and $s_i'(x)$ being the single–object probability of existence and probability density. Then, given the SPS measurement $z$, noise likelihood $f_w(z)$ and probability of visibility $p_V(x)$ the corrected set of Bernoulli components are

$$\left\{(q_i(z), s_i(x \mid z))_{i=1}^{\nu'}\right\}$$

determined by

$$q_i(z) = \frac{\left(q_i' - q_i' \, s_i'[p_V]\right) \varphi^{\neq i}(z) + q_i' \, s_i'\left[p_V \; \varphi_\eta^{\neq i}\right](z)}{\left(1 - q_i' \, s_i'[p_V]\right)\varphi^{\neq i}(z) + q_i' s_i'\left[p_V \; \varphi_\eta^{\neq i}\right](z)}$$

$$s_i(x \mid z) = \frac{(1 - p_V(x)) \, s_i'(x) \, \varphi^{\neq i}(z) + p_V(x) \, \varphi^{\neq i}(z - \eta(x)) \, s_i'(x)}{\left(1 - s_i'[p_V]\right) \varphi^{\neq i}(z) + s_i'\left[p_V(x) \, \varphi_\eta^{\neq i}\right](z)}$$

with $\varphi_\eta^{\neq i}(z \mid x) \triangleq \varphi^{\neq i}(z - \eta(x))$ and $s_i'[h] = \int h(x) \, s_i'(x) \, \mathrm{d}x$. Here,

$$\varphi^{\neq i}(z) \triangleq (f_w * \zeta_1[1] * \ldots * \zeta_{\neq i}[1] * \ldots * \zeta_{\nu'}[1])(z)$$

with

$$\zeta_i[1](z) \triangleq \left(1 - q_i' \, s_i'[p_V]\right)\delta_0(z) + q_i' s_i'\left[p_V \; \delta_\eta\right](z)$$

is the same Pseudo–likelihood as stated in Theorem 3 and therefore the same problems arise in computing it.

In contrast to the Approximate $\Sigma$–MeMBer filter, there is no distinction between missed and detected components. Thus, the number of components does not change and consequently is static.

#### 7.2.3.2. Derivation of the Intensity $\Sigma$–MeMBer Corrector

In this section, the results presented in Theorem 4 are proven. Therefore, first the PHD of the SPS–type MeMBer corrector is determined. Based on those results, approximate parameters for the resulting Bernoulli components will be derived.

*Proof of Theorem 4.* Recall the PGFL variant $G[h \mid Z_{1:k}]$ of the $\Sigma$–MeMBer filter corrector as stated in Theorem 1

$$G[h \mid Z_{1:k}] = \frac{(f_w * \zeta_1[h] * \ldots * \zeta_{\nu'}[h])(z)}{(f_w * \zeta_1[1] * \ldots * \zeta_{\nu'}[1])(z)}$$

with

$$\zeta_i[h](z) \triangleq \left(1 - q_i' + q_i' \, s_i'[h \, (1 - p_V)]\right)\delta_0(z) + q_i' s_i'\left[h \, p_V \; \delta_\eta\right](z)$$

and

$$\zeta_i[1](z) \triangleq \left(1 - q_i' \, s_i'[p_V]\right)\delta_0(z) + q_i' s_i'\left[p_V \; \delta_\eta\right](z).$$

Applying the methodologies of finite set statistics (FISST), the corresponding PHD becomes

$$\begin{aligned} D(x \mid Z_{1:k}) &= \frac{\delta G}{\delta x}[1 \mid Z_{1:k}] \\ &= K^{-1} \sum_{j=1}^{\nu'} \left( f_w * \zeta_1[1] * \ldots * \frac{\delta \zeta_j}{\delta x}[1] * \ldots * \zeta_{\nu'}[1] \right)(z) \\ &= \sum_{j=1}^{\nu'} K^{-1} \left( \varphi^{\neq j} * \frac{\delta \zeta_j}{\delta x}[1] \right)(z) \\ &= \sum_{j=1}^{\nu'} D_j(x \mid Z_{1:k}), \end{aligned}$$

where

$$D_j(x \mid Z_{1:k}) = K^{-1} \left( \varphi^{\neq j} * \frac{\delta \zeta_j}{\delta x}[1] \right)(z)$$

with

$$\begin{aligned} K &\triangleq \left(f_w * \zeta_1[1] * \ldots * \zeta_{\nu'}[1]\right)(z) \\ &= \left(\varphi^{\neq j} * \zeta_j[1]\right)(z) \end{aligned}$$

being the Bayes normalization constant and

$$\varphi^{\neq j}(z) \triangleq \left(f_w * \zeta_1[1] * \ldots * \zeta_{j-1}[1] * \zeta_{j+1}[1] * \ldots * \zeta_{\nu'}[1]\right)(z)$$

being the convolution of all $\zeta_i[1](z)$ but not the $j$–th. After inserting the functional derivative back into $D_j(x \mid Z_{1:k})$ it follows that

$$\begin{aligned} D_j(x \mid Z_{1:k}) &= \frac{\left(\varphi^{\neq j} * \left(q_j' \, (1 - p_V(x)) \, s_j'(x) \, \delta_0 + q_j' \, p_V(x) \, \delta_{\eta(x)} \, s_j'(x)\right)\right)(z)}{\left(\varphi^{\neq j} * \left(\left(1 - q_j' \, s_j'[p_V]\right)\delta_0 + q_j' s_j'\left[p_V \; \delta_\eta\right]\right)\right)(z)} \\ &= \frac{q_j' \, (1 - p_V(x)) \, s_j'(x) \, \varphi^{\neq j}(z) + q_j' \, p_V(x) \, \varphi^{\neq j}(z - \eta(x)) \, s_j'(x)}{\left(1 - q_j' \, s_j'[p_V]\right)\varphi^{\neq j}(z) + q_j' s_j'\left[p_V \; \varphi_\eta^{\neq j}\right](z)} \end{aligned}$$

where $\varphi_\eta^{\neq j}(z \mid x) \triangleq \varphi^{\neq j}(z - \eta(x))$.

Given the single–component PHD $D_j(x \mid Z_{1:k})$, then the approximate Bernoulli parameters $q_j$ and $s_j(x)$ can easily be derived. From sections 3.3 and sections 3.4

it is known that the PHD of a single Bernoulli component is $D_j(x) = q_j\, s_j(x)$. Thus, it is possible to derive approximate Bernoulli parameters from the PHD and as such it follows that

$$q_j(z) \approx \int D_j(x \mid Z_{1:k})\, \mathrm{d}x$$

and

$$\begin{aligned} s_j(x \mid z) &\approx \frac{D_j(x \mid Z_{1:k})}{\int D_j(x \mid Z_{1:k})\, \mathrm{d}x} \\ &= \frac{D_j(x \mid Z_{1:k})}{q_j(z)}. \end{aligned}$$

Based on these insights it is now possible to derive the intensity approximation of the Bernoulli parameters. The expected number of objects, which is in this case the probability of existence $q_j$, can be deduced by integrating over the PHD and as such becomes

$$\begin{aligned} q_j(z) &= \int D_j(x \mid Z_{1:k})\, \mathrm{d}x \\ &= \frac{\left(q_j' - q_j'\, s_j'[p_V]\right)\, \varphi^{\neq j}(z) + q_j'\, s_j'\left[p_V\, \varphi_\eta^{\neq j}\right](z)}{\left(1 - q_j'\, s_j'[p_V]\right) \varphi^{\neq j}(z) + q_j' s_j'\left[p_V\, \varphi_\eta^{\neq j}\right](z)}. \end{aligned}$$

Similarly to the proof of the probability of existence in the previous paragraph, the intensity approximation of the components PDFs follows directly from the resulting *PHD*.

The PDF is approximately

$$\begin{aligned} s_j(x) &\approx q_j^{-1}(z)\, D_j(x \mid Z_{1:k}) \\ &= \frac{\frac{q_j'\,(1-p_V(x))\, s_j'(x)\, \varphi^{\neq j}(z) + q_j'\, p_V(x)\, \varphi^{\neq j}(z-\eta(x))\, s_j'(x)}{\cancel{\left(1-q_j'\, s_j'[p_V]\right)\varphi^{\neq j}(z) + q_j' s_j'\left[p_V\, \varphi_\eta^{\neq j}\right](z)}}}{\frac{\left(q_j' - q_j'\, s_j'[p_V]\right)\, \varphi^{\neq j}(z) + q_j'\, s_j'\left[p_V\, \varphi_\eta^{\neq j}\right](z)}{\cancel{\left(1-q_j'\, s_j'[p_V]\right)\varphi^{\neq j}(z) + q_j' s_j'\left[p_V\, \varphi_\eta^{\neq j}\right](z)}}} \\ &= \frac{q_j'\,(1-p_V(x))\, s_j'(x)\, \varphi^{\neq j}(z) + q_j'\, p_V(x)\, \varphi^{\neq j}(z-\eta(x))\, s_j'(x)}{\left(q_j' - q_j'\, s_j'[p_V]\right)\, \varphi^{\neq j}(z) + q_j'\, s_j'\left[p_V(x)\, \varphi_\eta^{\neq j}\right](z)} \\ &= \frac{(1-p_V(x))\, s_j'(x)\, \varphi^{\neq j}(z) + p_V(x)\, \varphi^{\neq j}(z-\eta(x))\, s_j'(x)}{s_j'[1-p_V]\, \varphi^{\neq j}(z) + s_j'\left[p_V(x)\, \varphi_\eta^{\neq j}\right](z)} \\ &= \frac{L_j(z-\eta(x))\, s_j'(x)}{\int L_j(z-\eta(x))\, s_j'(x)\, \mathrm{d}x} \end{aligned}$$

with

$$L_j(z-\eta(x)) \triangleq (1-p_V(x))\,\varphi^{\neq j}(z) + p_V(x)\,\varphi^{\neq j}(z-\eta(x)).$$

Note that $L_j(z-\eta(x))$ is a valid likelihood and as such this concludes the proof. □

# 8. Multi–Bernoulli Pseudo–Likelihoods

## Contents



In Section 7.2, the principal equations for the superpositional multi–object multi–Bernoulli ($\Sigma$–MeMBer) filters have been presented. However, due to the computational complexity which originates from the convolutions necessary to compute the Pseudo–likelihood $\varphi^{\neq i}(z)$, those equations are still computationally intractable.

In this chapter, three computationally tractable approximations for the $\Sigma$–MeMBer filter equations are provided. In order to derive these approximations, it is made use of the fact that the main computational complexity originates from the determination of the Pseudo–likelihood $\varphi^{\neq i}(z)$ as seen in Theorems 3 and 4.

## 8.1. Exact Pseudo–Likelihood

Recalling from Theorems 3 and 4, that the Pseudo–likelihood is

$$\varphi^{\neq i}(z) \triangleq (f_w * \zeta_1[1] * \ldots * \zeta_{i-1}[1] * \zeta_{i+1}[1] * \ldots * \zeta_{\nu'}[1])(z)$$

with

$$\zeta_i[1](z) \triangleq \left(1 - q_i' \, s_i'[p_V]\right)\delta_0(z) + q_i' s_i' \left[p_V \; \delta_\eta\right](z),$$

then it can be noted that its complexity is hidden behind the convolution. Furthermore, by noticing that the Pseudo–likelihood is basically the convolution of

all $\zeta_i[1](z)$ but not the $j$–th element, where each $\zeta_i[1](z)$ depends solely on the $i$–th Bernoulli component $\left(q_i', s_i'(x)\right)$, it is possible to define the Pseudo–likelihood on a modified set of Bernoulli parameters

$$\{(q_i, s_i(x))_{i=1}^{\nu}\} = \left\{\left(q_i', s_i'(x)\right)_{i=1}^{\nu'}\right\} \setminus \left\{\left(q_j', s_j'(x)\right)\right\}.$$

Then, the problem of finding an appropriate approximation for $\varphi^{\neq j}(z)$ can be simplified to finding an appropriate approximation for

$$\varphi(z) = (f_w * \zeta_1[1] * \ldots * \zeta_\nu[1])(z) \tag{8.1}$$

with

$$\zeta_i[1](z) \triangleq (1 - q_i\, s_i[p_V])\, \delta_0(z) + q_i\, s_i\left[p_V\ \delta_\eta\right](z)$$

and

$$\nu = \nu' - 1$$

where $\varphi(z)$ is the Pseudo–likelihood considering the components $\{(q_i, s_i(x))_{i=1}^{\nu}\}$. Furthermore, the Pseudo–likelihood can be reformulated to

$$\begin{aligned}
\zeta_i[1](z) &= (1 - q_i\, s_i[p_V])\, \delta_0(z) + q_i\, s_i\left[p_V\ \delta_\eta\right](z) \\
&= (1 - q_i\, s_i[p_V])\, \delta_0(z) + q_i \int p_V(x)\, \delta_{\eta(x)}(z)\, s_i(x)\, \mathrm{d}x \\
&= (1 - q_i\, s_i[p_V])\, \delta_0(z) + q_i\, s_i[p_V] \int \frac{p_V(x)\, \delta_{\eta(x)}(z)\, s_i(x)}{s_i[p_V]}\, \mathrm{d}x.
\end{aligned}$$

Which, when substituting

$$\hat{q}_i = q_i\, s_i[p_V] \tag{8.2}$$

and

$$\hat{s}_i(x) = \frac{p_V(x)\, s_i(x)}{s_i[p_V]} \tag{8.3}$$

finally simplifies to

$$\begin{aligned}
\zeta_i[1](z) &= (1 - \hat{q}_i)\, \delta_0(z) + \hat{q}_i \int \delta_{\eta(x)}(z)\, \hat{s}_i(x)\, \mathrm{d}x \\
&= (1 - \hat{q}_i)\, \delta_0(z) + \hat{q}_i\, \hat{s}_i\left[\delta_\eta\right](z).
\end{aligned}$$

Having a closer look at those equations, then it is recognizable that

$$\hat{s}_i\left[\delta_\eta\right](z) = \int \delta_{\eta(x)}(z)\, \hat{s}_i(x)\, \mathrm{d}x$$

is the transformation of the random variable $\boldsymbol{x}$ with probability density function (PDF) $\hat{s}_i(x)$ from the single–object state space $\mathbb{X}$ to the single–measurement space $\mathbb{Z}$. Consequently, it is evident that

$$\zeta_i[1](z) = (1 - \hat{q}_i)\,\delta_0(z) + \hat{q}_i\; \hat{s}_i\left[\delta_\eta\right](z) \tag{8.4}$$

is essentially the PDF of a distribution in measurement space. More specifically, it is a mixture of a Dirac delta density $\delta_0(z)$ concentrated at $z = 0$ and the density $\hat{s}_i\left[\delta_\eta\right](z)$ with mixture weights $1 - \hat{q}_i$ and $\hat{q}_i$.

## 8.2. Quasi–Gaussian Pseudo–Likelihood

Since all $\zeta_i[1](z)$ are valid PDFs the objective is to find a replacement PDF that resembles $\zeta_i[1](z)$ as closely as possible but leads to a more computationally efficient Pseudo–likelihood. It is a common approach to approximate a distribution by matching the moments of $\zeta_i[1](z)$ as close as possible. Here, a Gaussian approximation is presented. Thus, $\zeta_i[1](z)$ is chosen to be approximated by its first and second central moments.

### 8.2.1. Main Results of the QG Pseudo–Likelihood

In the following the main results will be provided. It can be shown that the mean and the variance of $\zeta_i[1](z)$ are as follows.

**Proposition 19.** Given the transformed PDF $\zeta_i[1](z)$ with Bernoulli parameters $(q_i, s_i(x))$, the measurement function $\eta(x)$ and probability of visibility $p_V(x)$, then its mean $m_{\zeta_i}$ and variance $P_{\zeta_i}$ are

$$m_{\zeta_i} = q_i\; s_i[\eta\; p_V]$$

and

$$P_{\zeta_i} = q_i\; s_i[\eta\eta^\intercal\; p_V] - m_{\zeta_i} m_{\zeta_i}^\intercal$$

with

$$s_i[h] \triangleq \int h\; s_i(x)\; \mathrm{d}x.$$

*Proof.* See Section 8.2.2.

Given the results from Proposition 19, it is possible to approximate $\zeta_i(z)$ as a Normal distribution $\mathcal{N}_{m_{\zeta_i}, P_{\zeta_i}}(z)$.

**Definition 2.** Let $m_{\zeta_i}$ and $P_{\zeta_i}$ be the mean and variance of the distribution $\zeta_i[1](z)$, then the Gaussian approximation becomes

$$\zeta_i[1](z) \approx \mathcal{N}_{m_{\zeta_i}, P_{\zeta_i}}(z)$$

where

$$\mathcal{N}_{m,P}(z) \triangleq (2\,\pi)^{-\frac{k}{2}}\ |P|^{-\frac{1}{2}}\ \mathrm{e}^{-\frac{1}{2}\,(z-m)^{\intercal}\,P^{-1}\,(z-m)}$$

denotes the PDF of a multivariate normal or Gaussian distribution with location parameter $m$ and scale parameter $P$ with $|P| \stackrel{\text{abbr}}{=} \det P$ being the matrix determinant and $P^{-1}$ being the inverse of matrix $P$.

**Definition 3.** Given Definition 2, then the Gaussian approximation of the Pseudo–likelihood, further denoted the Quasi–Gaussian Pseudo–likelihood, becomes

$$\varphi(z) \approx \left(f_w * \mathcal{N}_{m_{\zeta_1},P_{\zeta_1}} * \ldots * \mathcal{N}_{m_{\zeta_{\hat{\nu}}},P_{\zeta_{\hat{\nu}}}}\right)(z),$$

which can be greatly simplified since the convolution of two Normal PDFs is again a Normal PDF with

$$\left(\mathcal{N}_{m_1,P_1} * \mathcal{N}_{m_2,P_2}\right)(z) = \mathcal{N}_{m_1+m_2,P_1+P_2}(z) \tag{8.5}$$

and therefore the Quasi–Gaussian Pseudo–likelihood simplifies as stated in Theorem 5.

**Theorem 5.** Given the Gaussian approximation of $\zeta_i[1](z)$ from Definition 2 which is specified by its means $m_{\zeta_j}$ and covariances $P_{\zeta_j}$ as determined by Proposition 19 and Definition 3, then the Quasi–Gaussian Pseudo–likelihood becomes

$$\varphi(z) = \left(f_w * \mathcal{N}_{m_\zeta,P_\zeta}\right)(z)$$

with

$$m_\zeta \triangleq \sum_{j=1}^{\nu} m_{\zeta_j}$$

and

$$P_\zeta \triangleq \sum_{j=1}^{\nu} P_{\zeta_j}.$$

*Proof.* The proof follows directly from Proposition 19 and Definitions 2 and 3. □

*Remark* 2. Considering that the noise PDF is Gaussian distributed with $f_w(z) = \mathcal{N}(z; m_w, P_w)$, then Quasi–Gaussian Pseudo–likelihood becomes

$$\varphi(z) = \mathcal{N}_{m_\zeta+m_w,P_\zeta+P_w}(z).$$

Summing it up, Theorem 5 provides a Gaussian approximation to the full Pseudo–likelihood with much lower computational complexity. However, the approximation is rather crude and much information is lost. The question remains, if there is a trade–off between the full Pseudo–likelihood and the Quasi–Gaussian Pseudo–likelihood.

### 8.2.2. Derivation of the QG Pseudo–Likelihood

In this section, the results from Section 8.2 are proven. Therefore, the first two central moments of the PDF $\zeta_i[1](z)$ are derived.

*Proof of Proposition 19.* Given the PDF $\zeta_i[1](z)$, then the first moment, the mean, becomes

$$\begin{aligned} m_{\zeta_i} &\triangleq \mathbb{E}_{\zeta_i}[z] \\ &= \int z\, \zeta_i[1](z)\, \mathrm{d}z \\ &= \int z \left( (1-\hat{q}_i)\, \delta_0(z) + \hat{q}_i\, \hat{s}_i\left[\delta_\eta\right](z) \right) \mathrm{d}z \\ &= \underbrace{\int z\, (1-\hat{q}_i)\, \delta_0(z)\, \mathrm{d}z}_{=0} + \int z\, \hat{q}_i\, \hat{s}_i\left[\delta_\eta\right](z)\, \mathrm{d}z \\ &= \int z\, \hat{q}_i\, \hat{s}_i\left[\delta_\eta\right](z)\, \mathrm{d}z. \end{aligned}$$

Further, since $\hat{s}_i[\delta_\eta](z) = \int \delta_{\eta(x)}(z)\, \hat{s}_i(x)\, \mathrm{d}x$ this becomes

$$m_{\zeta_i} = \hat{q}_i \int z \int \delta_{\eta(x)}(z)\, \hat{s}_i(x)\, \mathrm{d}x\, \mathrm{d}z.$$

Changing the order of integration finally results in

$$\begin{aligned} m_{\zeta_i} &= \hat{q}_i \int \int z\, \delta_{\eta(x)}(z)\, \mathrm{d}z\, \hat{s}_i(x)\, \mathrm{d}x \\ &= \hat{q}_i \int \eta(x)\, \hat{s}_i(x)\ \mathrm{d}x. \end{aligned}$$

Replacing $\hat{q}_i$ and $\hat{s}_i(x)$ by its definitions from (8.2) and (8.3) gives the proposed result

$$\begin{aligned} m_{\zeta_i} &= q_i \cancel{s_i[p_V]} \int \eta(x)\, \frac{p_V(x)\, s_i(x)}{\cancel{s_i[p_V]}}\, \mathrm{d}x \\ &= q_i \int \eta(x)\, p_V(x)\, s_i(x)\, \mathrm{d}x \\ &= q_i\, s_i[\eta\, p_V]. \end{aligned}$$

Similarly, the second central moment, the variance, becomes

$$\begin{aligned} P_{\zeta_i} &= \mathbb{E}_{\zeta_i}[zz^\intercal] - \mathbb{E}_{\zeta_i}[z]\, \mathbb{E}_{\zeta_i}[z]^\intercal \\ &= M_{\zeta_i} - m_{\zeta_i} m_{\zeta_i}^\intercal. \end{aligned}$$

Here, $M_{\zeta_i}$ is the second raw moment defined as

$$\begin{aligned} M_{\zeta_i} &= \mathbb{E}_{\zeta_i}[zz^\intercal] \\ &= q_i \int zz^\intercal \, \zeta_i[1](z) \, \mathrm{d}z \\ &= \int zz^\intercal \left( (1-\hat{q}_i)\, \delta_0(z) + \hat{q}_i \, \hat{s}_i\left[\delta_\eta\right](z) \right) \mathrm{d}z \\ &= \underbrace{\int zz^\intercal \, (1-\hat{q}_i)\, \delta_0(z) \, \mathrm{d}z}_{=0} + \int zz^\intercal \, \hat{q}_i \, \hat{s}_i\left[\delta_\eta\right](z) \, \mathrm{d}z \\ &= \int zz^\intercal \, \hat{q}_i \, \hat{s}_i\left[\delta_\eta\right](z) \, \mathrm{d}z. \end{aligned}$$

After expanding $\hat{s}_i\left[\delta_\eta\right](z) = \int \delta_{\eta(x)}(z) \, \hat{s}_i(x) \, \mathrm{d}x$, it follows that

$$\begin{aligned} M_{\zeta_i} &= \hat{q}_i \int zz^\intercal \int \delta_{\eta(x)}(z) \, \hat{s}_i(x) \, \mathrm{d}x \, \mathrm{d}z \\ &= \hat{q}_i \int zz^\intercal \, \delta_{\eta(x)}(z) \, \mathrm{d}z \int \hat{s}_i(x) \, \mathrm{d}x \\ &= \hat{q}_i \int \eta(x)\eta(x)^\intercal \, \hat{s}_i(x) \, \mathrm{d}x. \end{aligned}$$

Replacing $\hat{q}_i$ and $\hat{s}_i(x)$ by its definitions from (8.2) and (8.3) yields

$$\begin{aligned} M_{\zeta_i} &= q_i \int \eta(x)\eta(x)^\intercal \, p_V(x) \, s_i(x) \, \mathrm{d}x \\ &= q_i \, s_i[\eta\eta^\intercal \, p_V] \, \mathrm{d}x. \end{aligned}$$

As proposed, the variance finally becomes

$$\begin{aligned} P_{\zeta_i} &= M_{\zeta_i} - m_{\zeta_i} m_{\zeta_i}^\intercal \\ &= q_i \, s_i[\eta\eta^\intercal p_V] - q_i^2 \, s_i[\eta p_V] \, s_i[\eta p_V]^\intercal. \end{aligned}$$

□

## 8.3. Quasi–Gaussian Mixture Pseudo–Likelihood

In Section 8.2 a computationally tractable alternative to the full superpositional (SPS)–type multi–object multi–Bernoulli (MeMBer) filter has been proposed. While the applied approximation allows an efficient computation, the approximation is rather crude and much information is lost.

### 8.3.1. Main Results of the QGM Pseudo–Likelihood

In Chapter 8 it was pointed out that the Pseudo–likelihood $\varphi(z)$ is comprised out of the convolution of individual distributions with PDFs $\zeta_i[1](z) = (1-\hat{q}_i)\delta_0(z) + \hat{q}_i\ \hat{s}_i\left[\delta_\eta\right](z)$ (see (8.4)). It was chosen to approximate $\zeta_i[1](z) \approx \mathcal{N}_{m,P}(z)$ which consequently lead to the Quasi–Gaussian approximation stated in Theorem 5.

Here, another idea is pursued. Considering, that the Pseudo–likelihood

$$\varphi(z) = (f_w * \zeta_1[1] * \ldots * \zeta_\nu[1])(z)$$

with

$$\begin{aligned}\zeta_i[1](z) &= (1-\hat{q}_i)\delta_0(z) + \hat{q}_i\ \hat{s}_i\left[\delta_\eta\right](z)\\ &= (1-\hat{q}_i)\left(\delta_0(z) + \frac{\hat{q}_i\ \hat{s}_i\left[\delta_\eta\right](z)}{(1-\hat{q}_i)}\right)\end{aligned}$$

can be rewritten to

$$\varphi(z) = \prod_{i=1}^{\nu}(1-\hat{q}_i)\ f_w(z) * \left(\delta_0 + \frac{\hat{q}_1\ \hat{s}_1\left[\delta_\eta\right]}{(1-\hat{q}_1)}\right) * \cdots * \left(\delta_0 + \frac{\hat{q}_\nu\ \hat{s}_\nu\left[\delta_\eta\right]}{(1-\hat{q}_\nu)}\right)(z)$$

Then, after applying the identity of the convoluted elementary symmetric function (CESF) from (4.15) this expands to the following combinatorial sum

$$\varphi(z) = \prod_{i=1}^{\nu}(1-\hat{q}_i)\left(f_w * \sum_{n=0}^{\nu}\sum_{1\le i_1<\ldots<i_n\le\nu}\frac{\hat{q}_{i_1}\ \hat{s}_{i_1}\left[\delta_\eta\right]}{1-\hat{q}_{i_1}} * \ldots * \frac{\hat{q}_{i_n}\ \hat{s}_{i_n}\left[\delta_\eta\right]}{1-\hat{q}_{i_n}}\right)(z). \quad (8.6)$$

Hence, a Gaussian mixture (GM) approximation as stated in Theorem 6 can easily be established when assuming that the PDF is approximately Gaussian with $\hat{s}_i\left[\delta_\eta\right](z) \approx \mathcal{N}_{m_{\hat{s}_i},P_{\hat{s}_i}}(z)$. The parameters of the Normal distribution are then determined as described subsequently.

**Proposition 20.** Given the measurement function $z = \eta(x)$, probability of visibility $p_V(x)$ and Bernoulli parameters $(q_i, s_i(x))$, then the mean and variance of $\hat{s}_i\left[\delta_\eta\right](z)$ become

$$m_{\hat{s}_i} = \frac{s_i[\eta p_V]}{s_i[p_V]}$$

and

$$P_{\hat{s}_i} = \frac{s_i[\eta\eta^\intercal p_V]}{s_i[p_V]} - m_{\hat{s}_i}m_{\hat{s}_i}^\intercal$$

where $s_i[h] \triangleq \int h(x)\ s_i(x)\ \mathrm{d}x$.

*Proof.* See Section 8.3.2.

Replacing $\hat{s}_i\left[\delta_\eta\right](z)$ by its Gaussian approximation in (8.6) leads to the following approximation to the exact Pseudo–likelihood which is a direct result of the Gaussian convolution identity from (8.5).

**Theorem 6.** Let the PDF $\hat{s}_i\left[\delta_\eta\right](z)$ be approximately Gaussian $\mathcal{N}_{m_{\hat{s}_i},P_{\hat{s}_i}}(z)$ with mean $m_{\hat{s}_i}$ and covariance $P_{\hat{s}_i}$, then the Pseudo–likelihood becomes

$$\varphi(z) \approx \sum_{n=0}^{\nu} \sum_{1\leq i_1<\ldots<i_n\leq\nu} w_{\hat{s}_{i_{1:n}}} \left(f_w * \mathcal{N}_{m_{\hat{s}_{i_{1:n}}},P_{\hat{s}_{i_{1:n}}}}\right)(z)$$

where

$$\begin{aligned}
w_{\hat{s}_{i_{1:n}}} &= \prod_{i=1}^{\nu}(1-q_i\ s_i\left[p_V\right]) \left(\prod_{j=1}^{n}\frac{q_{i_j}\ s_{i_j}\left[p_V\right]}{1-q_{i_j}\ s_{i_j}\left[p_V\right]}\right) \\
m_{\hat{s}_{i_{1:n}}} &= \sum_{j=1}^{n} m_{\hat{s}_{i_j}} \\
P_{\hat{s}_{i_{1:n}}} &= \sum_{j=1}^{n} P_{\hat{s}_{i_j}}.
\end{aligned}$$

*Proof.* See Section 8.3.2.

Under the premise that the noise PDF $f_w(z)$ is also a Gaussian, then the approximate Pseudo–likelihood is a mixture of Gaussians (MoG) due to the convolution identity of Gaussians as described in (8.5).

*Remark* 3. Considering that the noise PDF is Gaussian distributed $f_w(z) = \mathcal{N}_{m_w,P_w}(z)$, then the approximate Pseudo–likelihood is a GM with PDF

$$\varphi(z) \approx \sum_{n=0}^{\nu} \sum_{1\leq i_1<\ldots<i_n\leq\nu} w_{\hat{s}_{i_{1:n}}}\ \mathcal{N}_{m_{\hat{s}_{i_{1:n}}}+m_w,P_{\hat{s}_{i_{1:n}}}+P_w}(z)$$

with

$$\sum_{n=0}^{\nu} \sum_{1\leq i_1<\ldots<i_n\leq\nu} w_{\hat{s}_{i_{1:n}}} = 1.$$

Having a closer look at Theorem 6 reveals that still all $2^\nu$ combinations have to be generated. However, the convolution is replaced by simply summing the means and variances. Therefore, if $s_i(x)$ is a complex PDF, such as a mixture of Dirac delta densities as employed in Sequential Monte Carlo (SMC) methods or Particle filters (PFs), then it will reduce the computational demand.

### 8.3.2. Derivation of the QGM Pseudo–Likelihood

Since Theorem 6 is a direct result of Proposition 20, it will be proven next that the mean and variance as presented in the proposition are as stated.

*Proof of the Proposition 20.* By definition the mean $m_{\hat{s}_i}$ of $\hat{s}_i\left[\delta_\eta\right](z)$ is

$$\begin{aligned} m_{\hat{s}_i} &= \mathbb{E}_{\hat{s}_i}[z] \\ &= \int z\, \hat{s}_i\left[\delta_\eta\right](z)\mathrm{d}z. \end{aligned}$$

Thus, with $\hat{s}_i\left[\delta_\eta\right](z) = \int \delta_{\eta(x)}(z)\, \hat{s}_i(x)\, \mathrm{d}x$ it becomes

$$\begin{aligned} m_{\hat{s}_i} &= \int z \int \delta_{\eta(x)}(z)\, \hat{s}_i(x)\, \mathrm{d}x\, \mathrm{d}z \\ &= \int\int z\, \delta_{\eta(x)}(z)\, \mathrm{d}z\, \hat{s}_i(x)\, \mathrm{d}x \\ &= \int \eta(x)\, \hat{s}_i(x)\, \mathrm{d}x. \end{aligned}$$

After substituting $\hat{s}_i(x) = \frac{p_V(x)\, s_i(x)}{s_i[p_V]}$ (see. (8.3)) and $s_i[h] \triangleq \int h(x)\, s_i(x)\, \mathrm{d}x$ the mean becomes as claimed

$$\begin{aligned} m_{\hat{s}_i} &= \frac{\int \eta(x)\, p_V(x)\, s_i(x)\, \mathrm{d}x}{s_i[p_V]} \\ &= \frac{s_i[\eta\, p_V]}{s_i[p_V]}. \end{aligned}$$

Similarly, the second raw moment of $\hat{s}_i\left[\delta_\eta\right](z)$ is

$$\begin{aligned} M_{\hat{s}_i} &= \mathbb{E}_{\hat{s}_i}[zz^\intercal] \\ &= \int\int zz^\intercal\, \hat{s}_i\left[\delta_\eta\right](z)\, \mathrm{d}z \\ &= \int\int zz^\intercal\, \delta_{\eta(x)}(z)\, \mathrm{d}z\, \hat{s}_i(x)\, \mathrm{d}x \\ &= \int \eta(x)\eta(x)^\intercal\, \hat{s}_i(x)\, \mathrm{d}x. \end{aligned}$$

Thus, after substituting $\hat{q}_i$ and $\hat{s}_i(x)$ by its definitions from (8.2) and (8.3), the variance becomes as claimed

$$\begin{aligned} P_{\hat{s}_i} &= M_{\hat{s}_i} - m_{\hat{s}_i} m_{\hat{s}_i}^\intercal \\ &= \frac{\int \eta(x)\eta(x)^\intercal\, p_V(x)\, s_i(x)\, \mathrm{d}x}{s_i[p_V]} - m_{\hat{s}_i} m_{\hat{s}_i}^\intercal \\ &= \frac{s_i[\eta\eta^\intercal\, p_V]}{s_i[p_V]} - \frac{s_i[\eta\, p_V] s_i[\eta\, p_V]^\intercal}{s_i[p_V]^2} \end{aligned}$$

□

After establishing the mean and the variance, it is now possible to proof Theorem 6.

*Proof of Theorem 6.* Under the assumption that all $\hat{s}_i\left[\delta_\eta\right](z)$ are approximately Gaussian with PDFs $\hat{s}_i(z) \approx \mathcal{N}_{m_{\hat{s}_i},P_{\hat{s}_i}}(z)$ where the corresponding means and variances are as described in Proposition 20, then the Pseudo–likelihood becomes

$$\begin{aligned}\varphi(z) &= \prod_{i=1}^{\nu}(1-\hat{q}_i)\left(f_w * \sum_{n=0}^{\nu}\sum_{1\le i_1<\ldots<i_n\le\nu}\frac{\hat{q}_{i_1}\,\hat{s}_{i_1}\left[\delta_\eta\right]}{1-\hat{q}_{i_1}} * \ldots * \frac{\hat{q}_{i_n}\,\hat{s}_{i_n}\left[\delta_\eta\right]}{1-\hat{q}_{i_n}}\right)(z)\\ &= \left(f_w * \sum_{n=0}^{\nu}\sum_{1\le i_1<\ldots<i_n\le\nu} w_{i_{1:n}}\left(\hat{s}_{i_1}\left[\delta_\eta\right] * \ldots * \hat{s}_{i_n}\left[\delta_\eta\right]\right)\right)(z)\\ &\approx \left(f_w * \sum_{n=0}^{\nu}\sum_{1\le i_1<\ldots<i_n\le\nu} w_{\hat{s}_{i_{1:n}}}\left(\mathcal{N}_{m_{\hat{s}_{i_1}},P_{\hat{s}_{i_1}}} * \ldots * \mathcal{N}_{m_{\hat{s}_{i_n}},P_{\hat{s}_{i_n}}}\right)\right)(z)\end{aligned}$$

with

$$w_{\hat{s}_{i_{1:n}}} \triangleq \prod_{i=1}^{\nu}(1-\hat{q}_i)\prod_{j=1}^{n}\frac{\hat{q}_{i_j}}{1-\hat{q}_{i_j}}.$$

After applying the Gaussian convolution identity

$$\left(\mathcal{N}_{m_1,P_1} * \mathcal{N}_{m_2,P_2}\right)(z) = \mathcal{N}_{m_1+m_2,P_1+P_2}(z),$$

it follows as claimed that

$$\varphi(z) = \sum_{n=0}^{\nu}\sum_{1\le i_1<\ldots<i_n\le\nu} w_{\hat{s}_{i_{1:n}}}\left(f_w * \mathcal{N}_{m_{\hat{s}_{i_{1:n}}},P_{\hat{s}_{i_{1:n}}}}\right)(z)$$

with

$$m_{\hat{s}_{i_{1:n}}} \triangleq \sum_{j=1}^{n} m_{\hat{s}_{i_j}}$$

and

$$P_{\hat{s}_{i_{1:n}}} \triangleq \sum_{j=1}^{n} P_{\hat{s}_{i_j}}.$$

This concludes the proof. □

## 8.4. Quasi–Poisson Binomial Pseudo–Likelihood

With the Quasi–Gaussian and the Gaussian Mixture Pseudo–likelihood presented in sections 8.2 and 8.3 two possible approximations to the full Pseudo–likelihood have been introduced. However, what is missing seems to be an intermediate step between those.

### 8.4.1. Main Results of the QPB Pseudo–Likelihood

Before presenting the main results, first consider the following reformulation of the Pseudo–likelihood.

**Proposition 21.** Given the Pseudo–likelihood $\varphi(z)$ as defined in (8.1), then another way of writing it is

$$\varphi(z) = (f_w * \zeta_{1:\nu})(z)$$

with

$$\zeta_{1:\nu}(z) = \sum_{n=0}^{\nu} \hat{p}(n)\, \zeta_{n,1:\nu}(z)$$

where

$$\zeta_{n,1:\nu}(z) \triangleq \frac{1}{\hat{p}(n)} \int \delta_{\eta(X)}(z)\, \delta_n(|X|)\, \hat{f}(X)\delta X$$

and

$$\hat{f}(X) = \prod_{i=1}^{\nu}(1-\hat{q}_i) \sum_{1\le i_1\neq\ldots\neq i_n\le\nu} \prod_{j=1}^{n} \frac{\hat{q}_{i_j}\, \hat{s}_{i_j}(x_j)}{1-\hat{q}_{i_j}}$$

$$\hat{p}(n) = \prod_{i=1}^{\nu}(1-\hat{q}_i) \sum_{1\le i_1<\ldots<i_n\le\nu} \prod_{j=1}^{n} \frac{\hat{q}_{i_j}}{1-\hat{q}_{i_j}}$$

is the PDF with corresponding probability mass function (PMF) for a multi–Bernoulli distribution with parameters $\{(\hat{q}_i, \hat{s}_i)_{i=1}^{\nu}\}$. Also, remember that $\hat{s}_i(x) = \frac{p_V(x)\, s_i(x)}{s_i[p_V]}$ and $\hat{q}_i = q_i\, s_i[p_V]$ with $s_i[p_V] = \int p_V(x)\, s_i(x)\, \mathrm{d}x$.

*Proof.* See Section 8.4.2.

Considering this, it should be possible to replace the multi–Bernoulli PDF with another multi–object PDF that allows a more efficient computation by finding an appropriate approximation of $\zeta_{n,1:\nu}(z)$. As before, it is chosen to approximate $\zeta_{n,1:\nu}(z)$ with a Gaussian $\mathcal{N}_{m_{\zeta_n}, P_{\zeta_n}}(z)$.

**Theorem 7.** Let $f_w(z)$ be the PDF of the additive noise and $\varphi(z)$ be formulated as described in Proposition 21, then the Poisson–Binomial approximation of the Pseudo–likelihood becomes

$$\varphi(z) \approx \sum_{n=0}^{\nu} p(n) \left( f_w * \mathcal{N}_{m_{\zeta_{n,1:\nu}}, P_{\zeta_{n,1:\nu}}} \right)(z)$$

with

$$m_{\zeta_{n,1:\nu}} = \frac{n \sum_{j=1}^{\nu} p^{\neq j}(n-1)\, q_j\, s_j[\eta\, p_V]}{\sum_{j=1}^{\nu} p^{\neq j}(n-1)\, q_j\, s_j[p_V]}$$

$$P_{\zeta_{n,1:\nu}} = \frac{n \sum_{j=1}^{\nu} p^{\neq j}(n-1)\, q_j\, s_j[\eta\eta^{\mathsf{T}}\, p_V]}{\sum_{j=1}^{\nu} p^{\neq j}(n-1)\, q_j\, s_j[p_V]} + \frac{n\,(n-1) \sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} p^{\neq j_1, j_2}(n-2)\, q_{j_1}\, q_{j_2}\, s_{j_1}[\eta\, p_V]\, s_{j_2}[\eta\, p_V]^{\mathsf{T}}}{\sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} p^{\neq j_1, j_2}(n-2)\, q_{j_1}\, q_{j_2}\, s_{j_1}[p_V]\, s_{j_2}[p_V]} - m_{\zeta_{n,1:\nu}} m_{\zeta_{n,1:\nu}}^{\mathsf{T}}$$

where $p^{\neq j}(n)$ and $p^{\neq j_1, j_2}(n)$ are the PMF of a Poisson Binomial distribution

$$p(n) = \prod_{i=1}^{\nu} (1 - q_i\, s_i[p_V]) \sum_{1 \leq i_1 < \ldots < i_n \leq \nu} \prod_{j=1}^{n} \frac{q_{i_j}\, s_i[p_V]}{1 - q_{i_j}\, s_i[p_V]}$$

which is the cardinality distribution of the multi–Bernoulli distribution as described in Section 3.4 with parameters $\mathcal{B} \triangleq \{q_1, \ldots, q_\nu\}$ but not the $j$–th or $j_1$–th and $j_2$–th Bernoulli component. Consequently, if $p(n) \overset{\text{abbr}}{=} p(n; \mathcal{B})$, then it follows that $p^{\neq j}(n) \overset{\text{abbr}}{=} p(n; \mathcal{B} \setminus \{q_j\})$ and $p^{\neq j_1, j_2}(n) \overset{\text{abbr}}{=} p(n; \mathcal{B} \setminus \{q_{j_1}, q_{j_2}\})$.

*Proof.* See Section 8.4.2.

Having a closer look at Theorem 7 reveals that the mean is basically the weighted sum of the expected mean of the individual component means scaled by probability $p^{\neq j}(n-1)$. The probability $p^{\neq j}(n)$ represents the probability that there are $n$ other objects present other then the $j$–th object. Unfortunately, the computation of $p^{\neq j}(n)$ is still computationally demanding as it is the PMF of a Poisson Binomial distribution with probabilities $\hat{q}_i = q_i\, s_i[p_V]$ as described in [Wan93]. This

publication also provides some useful identities and approximations to compute such PMFs recursively, which are also stated in Section 8.4.2. Also, recently Hong has shown in [Hon13] how to compute the PMF $p(n)$ of a Poisson Binomial distribution efficiently using the Discrete Fourier Transform (DFT) [Hon13, Alg. A].

*Remark* 4. Note that when the noise is Normal distributed with PDF $f_w(z) = \mathcal{N}_{m_w,P_w}(z)$ where $m_w$ and $P_w$ are the mean and covariance of the Normal distribution, then the result from Theorem 7 simplifies to

$$\varphi(z) \approx \sum_{n=0}^{\nu} p(n)\, \mathcal{N}_{m_{\zeta_{n,1:\nu}}+m_w,P_{\zeta_{n,1:\nu}}+P_w}(z)$$

*Proof.* The result follows directly from the Gaussian convolution identity as stated in (8.5). □

## 8.4.2. Derivation of the QPB Pseudo–Likelihood

In this section, the results from the previous Section 8.4.1 are derived. At first, it is necessary to establish some identities for the multi–Bernoulli PDF and its underlying cardinality distribution. Using these identities, the principal equations for calculating the mean and the variance for the Quasi–Poisson Binomial approximation of the Pseudo–likelihood are derived.

First recall that the multi–Bernoulli cardinality distribution of Section 3.4 is

$$\begin{aligned} p(n) &= \prod_{i=1}^{\nu}(1-q_i) \sum_{1\le i_1<\ldots<i_n\le\nu} \prod_{j=1}^{n} \frac{q_{i_j}}{1-q_{i_j}} \\ &= \prod_{i=1}^{\nu}(1-q_i)\, \sigma_{\nu,n}\left(\frac{q_1}{1-q_1},\ldots,\frac{q_\nu}{1-q_\nu}\right) \end{aligned}$$

and as such it is essentially the PMF $p(n) \overset{\text{abbr}}{=} p(n;\mathcal{B})$ of a Poisson Binomial distribution with set of parameters $\mathcal{B} \triangleq \{q_1,\ldots,q_\nu\}$. Therefore, regarding to [Wan93, Lemma 1] it satisfies the following identity.

**Proposition 22** (Poisson–Binomial Recursion)**.** Let $1 \le k \le n \le \nu$, then the PMF of a multi–Bernoulli random finite set (RFS) with $\nu$ Bernoulli components satisfies the recursion

$$p(n) = \frac{(n-k)!}{n!} \sum_{i_1=1}^{\nu} q_{i_1} \left( \sum_{\substack{i_2=1 \\ i_2\neq i_1}}^{\nu} q_{i_2} \cdots \left( \sum_{\substack{i_k=1 \\ i_k\neq i_1,\ldots,i_{k-1}}}^{\nu} q_{i_k}\, p^{\neq i_{1:k}}(n-k) \right) \right) \tag{8.7}$$

and $p^{\neq i_{1:k}}(n) \overset{\text{abbr}}{=} p\left(n;\mathcal{B}\setminus\left\{q_{i_1},\ldots,q_{i_k}\right\}\right)$ being the PMF of a Poisson Binomial distribution with parameters $\mathcal{B}\setminus\left\{q_{i_1},\ldots,q_{i_k}\right\}$.

*Proof.* See [Wan93, Lemma 1].

*Remark* 5 (Poisson–Binomial Recursion for $k = 1$ and $k = 2$). Let $k = 1$ and $k = 2$, then (8.7) can be simplified to

$$p(n) = \frac{1}{n} \sum_{j=1}^{\nu} q_j \, p^{\neq j}(n-1)$$

and

$$p(n) = \frac{1}{n(n-1)} \sum_{j_1=1}^{\nu} q_{j_1} \left( \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} q_{j_2} \, p^{\neq j_1, j_2}(k-2) \right).$$

While this is a common result in ordinary probability theory, as far as known to the author, there is no such result for the multi–object multi–Bernoulli PDFs in finite set statistics (FISST) or in Point Process Theory (PPT). However, it is possible to establish such.

**Proposition 23** (Multi–Object Multi–Bernoulli Recursion)**.** Let $f(X) \overset{\text{abbr}}{=} f(X; \mathcal{MB})$ be the PDF of a multi–Bernoulli RFS with set of parameters $\mathcal{MB} \triangleq \{(q_i, s_i(x))_{i=1}^{\nu}\}$ and finite set $X = \{x_1, \ldots, x_n\}$, then the PDF satisfies

$$\begin{aligned} f(X) = & \sum_{i_1=1}^{\nu} q_{i_1} \, s_{i_1}(x_1) \sum_{\substack{i_2=1 \\ i_2 \neq i_1}}^{\nu} q_{i_2} \, s_{i_2}(x_2) \\ & \cdots \sum_{\substack{i_k=1 \\ i_k \neq i_1, \ldots, i_{k-1}}}^{\nu} q_{i_k} \, s_{i_k}(x_k) \, f^{\neq i_{1:k}}(X \setminus \{x_1, \ldots, x_k\}) \\ = & \sum_{1 \leq i_1 \neq \ldots \neq i_k \leq \nu} \prod_{j=1}^{k} q_{i_j} \, s_{i_j}(x_j) \, f^{\neq i_{1:k}}(X \setminus \{x_1, \ldots, x_k\}) \end{aligned}$$

where $f^{\neq i_{1:k}}(X)$ is the PDF of a multi–Bernoulli RFS with set of $\mathcal{MB} \setminus \left\{ \left( q_{i_j}, s_{i_j}(x) \right)_{j=1}^{k} \right\}$.

*Proof.* Considering $\alpha_i(x) \triangleq \frac{q_i \, s_i(x)}{1 - q_i}$, $0 \leq k \leq n \leq \nu$ and $X_{=}\{x_1, \ldots, x_n\}$ , then it follows

that

$$\begin{aligned} f(X) &= \prod_{i=1}^{\nu}(1-q_i) \sum_{1\le i_1\neq\ldots\neq i_n\le\nu} \prod_{j=1}^{n} \alpha_{i_j}(x_j) \\ &= \prod_{i=1}^{\nu}(1-q_i) \sum_{i_1=1}^{\nu} \alpha_{i_1}(x_1) \sum_{\substack{i_2=1 \\ i_2\neq i_1}}^{\nu} \alpha_{i_2}(x_2) \\ &\qquad \cdots \sum_{\substack{i_k=1 \\ i_k\neq i_1,\ldots,i_k}}^{\nu} \alpha_{i_k}(x_k) \sum_{\substack{1\le i_{k+1}\neq\ldots\neq i_n\le\nu \\ i_l\neq i_m \\ l=1,\ldots,k \\ m=k+1,\ldots,n}} \prod_{j=k+1}^{n} \alpha_{i_j}(x_j). \end{aligned}$$

In turn, replacing $\alpha_1(x),\ldots,\alpha_\nu(x)$ by its definition leads to

$$\begin{aligned} f(X) = \sum_{i_1=1}^{\nu} q_{i_1}\, s_{i_1}(x_1) \sum_{\substack{i_2=1 \\ i_2\neq i_1}}^{\nu} q_{i_2}\, s_{i_2}(x_2) \\ \cdots \sum_{\substack{i_k=1 \\ i_k\neq i_1,\ldots,i_{k-1}}}^{\nu} q_{i_k}\, s_{i_k}(x_k) \underbrace{\frac{\prod_{i=1}^{\nu}(1-q_i)}{\prod_{i=1}^{k}(1-q_{i_k})} \sum_{\substack{1\le i_{k+1}\neq\ldots\neq i_n\le\nu \\ i_l\neq i_m \\ l=1,\ldots,k \\ m=k+1,\ldots,n}} \prod_{j=k+1}^{n} \frac{q_{i_j}\, s_{i_j}(x_j)}{1-q_{i_j}}}_{A}. \end{aligned}$$

Finally, realizing that term $A$ is in fact a multi–Bernoulli PDF $f^{\neq i_{1:k}}(X) \overset{\text{abbr}}{=} f(X; \mathcal{MB}\setminus \left\{\left(q_{i_j}, s_{i_j}(x)\right)_{j=1}^{k}\right\})$ with parameters $\mathcal{MB}\setminus\left\{\left(q_{i_j}, s_{i_j}(x)\right)_{j=1}^{k}\right\}$, then the equation simplifies to

$$\begin{aligned} f(X) &= \sum_{i_1=1}^{\nu} q_{i_1}\, s_{i_1}(x_1) \sum_{\substack{i_2=1 \\ i_2\neq i_1}}^{\nu} q_{i_2}\, s_{i_2}(x_2) \\ &\qquad \cdots \sum_{\substack{i_k=1 \\ i_k\neq i_1,\ldots,i_{k-1}}}^{\nu} q_{i_k}\, s_{i_k}(x_k)\, f^{\neq i_{1:k}}(X\setminus\{x_1,\ldots,x_k\}) \\ &= \sum_{1\le i_1\neq\ldots\neq i_k\le\nu} \prod_{j=1}^{k} q_{i_j}\, s_{i_j}(x_j)\, f^{\neq i_{1:k}}(X\setminus\{x_1,\ldots,x_k\}). \end{aligned} \tag{8.8}$$

□

*Remark* 6 (Multi–Object Multi–Bernoulli Recursion for $k = 1$ and $k = 2$). Let $k = 1$ and $k = 2$ then (8.8) reduces to

$$f(X) = \sum_{j=1}^{\nu} q_j \, s_j(x) \, f^{\neq j}(X \setminus \{x\})$$

and

$$f(X) = \sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_1 \neq j_2}}^{\nu} q_{j_1} \, s_{j_1}(x_1) \, q_{j_2} \, s_{j_2}(x_2) \, f^{\neq j_1, j_2}(X \setminus \{x_1, x_2\})$$

where $x, x_1, x_2 \in X$.

Based on the aforementioned propositions and remarks it is now possible to proof Proposition 21.

*Proof of Proposition 21.* First recall (8.1)

$$\varphi(z) = (f_w * \zeta_{1:\nu})(z)$$

with

$$\begin{aligned}\zeta_{1:\nu}(z) &\triangleq (\zeta_1[1] * \ldots * \zeta_\nu[1])(z) \\ &= \prod_{i=1}^{\nu}(1-\hat{q}_i) \sum_{n=0}^{\nu} \sum_{1 \leq i_1 < \ldots < i_n \leq \nu} \left( \frac{\hat{q}_{i_1} \, \hat{s}_{i_1}\left[\delta_\eta\right]}{1-\hat{q}_{i_1}} * \ldots * \frac{\hat{q}_{i_n} \, \hat{s}_{i_n}\left[\delta_\eta\right]}{1-\hat{q}_{i_n}} \right)(z).\end{aligned}$$

Since $\zeta_i(z)$ is the PDF of a random variable (RV) in the measurement space, the convolution of multiple $\zeta_{1:\nu}(z) = (\zeta_1[1] * \ldots * \zeta_\nu[1])(z)$ is also a valid PDF in measurement space. In essence, $\zeta_{1:n}(z)$ is the relative probability of the *sum of independent* RVs $\boldsymbol{z}_{1:\nu} = \sum_{i=1}^{\nu} \boldsymbol{z}_i$ where each $\boldsymbol{z}_i \sim \pi_{\zeta_i}$. Here, $\pi_{\zeta_i}$ is the distribution having PDF $\zeta_i(z)$.

The question arises if there is a reasonable approximation to the PDF $\zeta_{1:\nu}(z)$ that allows a finer control over the information loss. Reformulating the PDF reveals that

$$\zeta_{1:\nu}(z) = \sum_{n=0}^{\nu} \prod_{i=1}^{\nu}(1-\hat{q}_i) \sum_{1 \leq i_1 < \ldots < i_n \leq \nu} \left( \frac{\hat{q}_{i_1} \, \hat{s}_{i_1}\left[\delta_\eta\right]}{1-\hat{q}_{i_1}} * \ldots * \frac{\hat{q}_{i_n} \, \hat{s}_{i_n}\left[\delta_\eta\right]}{1-\hat{q}_{i_n}} \right)(z).$$

Further, expanding $\hat{s}_i\left[\delta_\eta\right](z) = \int \delta_\eta(x) \, \hat{s}_i(x) \, \mathrm{d}x$ shows that

$$\begin{aligned}\zeta_{1:\nu}(z) = \sum_{n=0}^{\nu} \frac{1}{n!} \int\!\cdots\!\int \delta\left(z - \sum_{x \in X} \eta(x)\right) \\ \prod_{i=1}^{\nu}(1-\hat{q}_i) \sum_{1 \leq i_1 \neq \ldots \neq i_n \leq \nu} \prod_{j=1}^{n} \frac{\hat{q}_{i_j} \, \hat{s}_{i_j}(x_j)}{1-\hat{q}_{i_j}} \, \mathrm{d}x_1 \cdots \mathrm{d}x_n\end{aligned}$$

which, by the definition of the set–integral, follows to be

$$\zeta_{1:\nu}(z) = \int \delta(z - \sum_{x \in X} \eta(x))\, \hat{f}(X)\, \delta X$$

where

$$\hat{f}(X) = \prod_{i=1}^{\nu} (1 - \hat{q}_i) \sum_{1 \leq i_1 \neq \ldots \neq i_n \leq \nu} \prod_{j=1}^{n} \frac{\hat{q}_{i_j}\, \hat{s}_{i_j}(x_j)}{1 - \hat{q}_{i_j}}$$

is the PDF of a multi–object multi–Bernoulli distribution as defined in Section 3.4.

Consider the multi–Bernoulli PDF $\hat{f}(X)$ and $\delta_{\eta(X)}(z) \overset{\text{abbr}}{=} \delta(z - \sum_{x \in X} \eta(X))$, the PDF can be reformulated as

$$\begin{aligned}
\zeta_{1:\nu}(z) &= \int \delta_{\eta(X)}(z)\, \hat{f}(X)\, \delta X \\
&= \sum_{n \geq 0} \int_{|X|=n} \delta_{\eta(X)}(z)\, \hat{f}(X)\, \delta X \\
&= \sum_{n=0}^{\nu} \int \delta_{\eta(X)}(z)\, \delta_n(|X|)\, \hat{f}(X)\, \delta X. \qquad (8.9)
\end{aligned}$$

In order to simplify (8.9), let

$$\hat{f}_n(X) \triangleq \frac{\delta_n(|X|)\, \hat{f}(X)}{\int \delta_n(|X|)\, \hat{f}(X)\, \delta X}$$

be the PDF that exactly $n$ objects are present. Additionally, realizing that the cardinality distribution $\hat{p}(n)$ of the multi–object distribution $\hat{f}(X)$ is

$$\begin{aligned}
\hat{p}(n) &= \int_{|X|=n} \hat{f}(X)\, \delta X \\
&= \int \delta_n(|X|)\, \hat{f}(X)\, \delta X,
\end{aligned}$$

then it follows that

$$\hat{f}_n(X) = \frac{1}{\hat{p}(n)}\, \delta_n(|X|)\, \hat{f}(X)$$

or equivalently

$$\hat{p}(n)\, \hat{f}_n(X) = \delta_n(|X|)\, \hat{f}(X).$$

Hence, (8.9) reduces to

$$\begin{aligned}\zeta_{1:\nu}(z) &= \sum_{n=0}^{\nu} \int \delta_{\eta(X)}(z)\, \hat{p}(n)\, \hat{f}_n(X)\delta X \\ &= \sum_{n=0}^{\nu} \hat{p}(n) \int \delta_{\eta(X)}(z)\, \hat{f}_n(X)\delta X.\end{aligned}$$

Finally, this results in

$$\zeta_{1:\nu}(z) = \sum_{n=0}^{\nu} \hat{p}(n)\, \zeta_{n,1:\nu}(z)$$

with

$$\begin{aligned}\zeta_{n,1:\nu}(z) &\triangleq \int \delta_{\eta(X)}(z)\, \hat{f}_n(X)\delta X \\ &= \int \delta_{\eta(X)}(z)\, \frac{1}{\hat{p}(n)}\, \delta_n(|X|)\, \hat{f}(X)\delta X \\ &= \frac{1}{\hat{p}(n)} \int \delta_{\eta(X)}(z)\, \delta_n(|X|)\, \hat{f}(X)\delta X.\end{aligned}$$

□

Given the alternative formulation of the Pseudo–likelihood, it is now possible to prove Theorem 7.

*Proof of Theorem 7.* After determining the PDF $\zeta_{k,1:\nu}(z)$, it is possible to derive its mean

$$\begin{aligned}m_{\zeta_{k,1:\nu}} &= \mathbb{E}_{\zeta_{k,1:\nu}}[z] \\ &= \int z\, \zeta_{k,1:\nu}(z)\, \mathrm{d}z \\ &= \frac{1}{\hat{p}(k)} \int z \int \delta_{\eta(X)}(z)\, \delta_k(|X|)\, \hat{f}(X)\, \delta X\, \mathrm{d}z \\ &= \frac{1}{\hat{p}(k)} \int \left( \int z\, \delta_{\eta(X)}(z)\, \mathrm{d}z \right) \delta_k(|X|)\, \hat{f}(X)\, \delta X \\ &= \frac{1}{\hat{p}(k)} \int \left( \sum_{x \in X} \eta(x) \right) \delta_k(|X|)\, \hat{f}(X)\, \delta X \\ &= \frac{1}{\hat{p}(k)} \frac{1}{k!} \int\cdots\int \left( \sum_{i=1}^{k} \eta(x_i) \right) \hat{f}\left(\{x_1, \ldots, x_k\}\right)\, \mathrm{d}x_1 \cdots \mathrm{d}x_k \\ &= \frac{1}{\hat{p}(k)} \frac{1}{k!} \int\cdots\int \sum_{i=1}^{k} \Big( \eta(x_i)\, \hat{f}\left(\{x_1, \ldots, x_k\}\right) \Big)\, \mathrm{d}x_1 \cdots \mathrm{d}x_k. \qquad (8.10)\end{aligned}$$

Furthermore, it is known from Proposition 23 that the multi–Bernoulli PDF $\hat{f}(X)$ can be expressed recursively. More specifically, Remark 6 provides the recursion formula for some special cases like $k = 1$. Therefore, $\hat{f}(X)$ can be expressed as

$$\hat{f}(X) = \sum_{j=1}^{\nu} \hat{q}_j \, \hat{s}_j(x) \, \hat{f}^{\neq j}(X \setminus \{x\}).$$

In turn, applying those results to (8.10) leads to

$$\begin{aligned} m_{\zeta_{k,1:\nu}} &= \frac{1}{\hat{p}(k)} \frac{1}{k!} \int\cdots\int \sum_{i=1}^{k} \eta(x_i) \sum_{j=1}^{\nu} \hat{q}_j \, \hat{s}_j(x_i) \, \hat{f}^{\neq j}(\{x_1,\ldots,x_k\} \setminus \{x_i\}) \, \mathrm{d}x_1 \cdots \mathrm{d}x_k \\ &= \frac{1}{\hat{p}(k)} \frac{1}{k!} \sum_{i=1}^{k} \sum_{j=1}^{\nu} \int\cdots\int \eta(x_i) \, \hat{q}_j \, \hat{s}_j(x_i) \, \hat{f}^{\neq j}(\{x_1,\ldots,x_k\} \setminus \{x_i\}) \, \mathrm{d}x_1 \cdots \mathrm{d}x_k \quad (8.11) \end{aligned}$$

Furthermore, with the definition of the set–integral from (2.2) the multivariate integral can be simplified to

$$\int\cdots\int \hat{f}^{\neq j}(\{x_1,\ldots,x_k\} \setminus \{x_i\}) \, \mathrm{d}x_1 \cdots \mathrm{d}x_k = \int (k-1)! \int_{|W_{\neq i}|=k-1} \hat{f}^{\neq j}(W_{\neq i}) \, \delta W_{\neq i} \, \mathrm{d}x_i$$

where $W_{\neq i} = X \setminus \{x_i\}$. Hence, (8.11) becomes

$$\begin{aligned} m_{\zeta_{k,1:\nu}} &= \frac{1}{\hat{p}(k)} \frac{1}{k!} \sum_{i=1}^{k} \sum_{j=1}^{\nu} \int \eta(x_i) \, \hat{q}_j \, \hat{s}_j(x_i) \, (k-1)! \int_{|W_{\neq i}|=k-1} \hat{f}^{\neq j}(W_{\neq i}) \, \delta W_{\neq i} \, \mathrm{d}x_i \\ &= \frac{1}{\hat{p}(k)} \frac{(k-1)!}{k!} \sum_{i=1}^{k} \sum_{j=1}^{\nu} \int \eta(x_i) \, \hat{q}_j \, \hat{s}_j(x_i) \int_{|W_{\neq i}|=k-1} \hat{f}^{\neq j}(W_{\neq i}) \, \delta W_{\neq i} \, \mathrm{d}x_i. \end{aligned}$$

Realizing that

$$\hat{p}^{\neq j}(k-1) = \int_{|W_{\neq i}|=k-1} \hat{f}^{\neq j}(W_{\neq i}) \, \delta W_{\neq i}$$

is the PMF of the multi–object PDF $\hat{f}^{\neq j}(W_{\neq i})$, the expression simplifies to

$$m_{\zeta_{k,1:\nu}} = \frac{1}{\hat{p}(k)} \frac{(k-1)!}{k!} \sum_{i=1}^{k} \sum_{j=1}^{\nu} \int \eta(x_i) \, \hat{q}_j \, \hat{s}_j(x_i) \, \hat{p}^{\neq j}(k-1) \, \mathrm{d}x_i.$$

Noticing that the result of the inner sum does not depend on a specific $x_i$ and

reordering the terms simplifies to

$$\begin{aligned} m_{\zeta_{k,1:\nu}} &= \frac{1}{\hat{p}(k)} \frac{(k-1)!}{k!} \sum_{i=1}^{k} \sum_{j=1}^{\nu} \int \eta(x)\, \hat{q}_j\, \hat{s}_j(x)\, \hat{p}^{\neq j}(k-1)\, \mathrm{d}x \\ &= \frac{1}{\hat{p}(k)} \frac{(k-1)!}{k!}\, k \sum_{j=1}^{\nu} p^{\neq j}(k-1)\, \hat{q}_j \int \eta(x)\, \hat{s}_j(x)\, \mathrm{d}x \\ &= \frac{1}{\hat{p}(k)} \cancel{\frac{(k-1)!}{k!}}\, k \sum_{j=1}^{\nu} p^{\neq j}(k-1)\, \hat{q}_j\, \hat{s}_j[\eta] \\ &= \frac{1}{\hat{p}(k)} \sum_{j=1}^{\nu} \hat{p}^{\neq j}(k-1)\, \hat{q}_j\, \hat{s}_j[\eta]. \end{aligned}$$

Finally applying Remark 5 results in

$$m_{\zeta_{k,1:\nu}} = \frac{k \sum_{j=1}^{\nu} \hat{p}^{\neq j}(k-1)\, \hat{q}_j\, \hat{s}_j[\eta]}{\sum_{j=1}^{\nu} \hat{p}^{\neq j}(k-1)\, \hat{q}_j}$$

and replacing $\hat{q}_j = q_j\, s_j[p_V]$ and $\hat{s}_j(x) = \frac{p_V(x)\, s_j(x)}{s_j[p_V]}$ gives the proposed result

$$m_{\zeta_{k,1:\nu}} = \frac{k \sum_{j=1}^{\nu} p^{\neq j}(k-1)\, q_j\, s_j[\eta\, p_V]}{\sum_{j=1}^{\nu} p^{\neq j}(k-1)\, q_j\, s_j[p_V]}.$$

Similarly to the previous proof the second raw moment of $\zeta_{k,1:\nu}(z)$ becomes

$$
\begin{aligned}
\mathbb{E}_{\zeta_{k,1:\nu}}[zz^{\intercal}] &= \int zz^{\intercal}\, \zeta_{k,1:\nu}(z)\, \mathrm{d}z \\
&= \frac{1}{\hat{p}(k)} \int \left(\sum_{x\in X} \eta(x)\right)\left(\sum_{x\in X} \eta(x)\right)^{\intercal} \delta_k(|X|)\, \hat{f}(X)\, \delta X \\
&= \frac{1}{\hat{p}(k)}\, \frac{1}{k!} \int\cdots\int \left(\sum_{i=1}^{k} \eta(x_i)\right)\left(\sum_{i=j}^{k} \eta(x_j)\right)^{\intercal} \hat{f}\left(\{x_1,\ldots,x_k\}\right)\, \mathrm{d}x_1 \cdots \mathrm{d}x_k \\
&= \frac{1}{\hat{p}(k)}\, \underbrace{\frac{1}{k!} \int\cdots\int \left(\sum_{i=1}^{k} \eta(x_i)\eta(x_i)^{\intercal}\right) \hat{f}\left(\{x_1,\ldots,x_k\}\right)\, \mathrm{d}x_1 \cdots \mathrm{d}x_k}_{A} \\
&\quad + \frac{1}{\hat{p}(k)}\, \underbrace{\frac{1}{k!} \int\cdots\int \left(\sum_{i_1=1}^{k} \sum_{\substack{i_2=1 \\ i_1\neq i_2}}^{k} \eta(x_{i_1})\eta(x_{i_2})^{\intercal}\right) \hat{f}\left(\{x_1,\ldots,x_k\}\right)\, \mathrm{d}x_1 \cdots \mathrm{d}x_k}_{B}.
\end{aligned}
$$

Similar to determining the mean, term $A$ becomes

$$
A \equiv \sum_{k=1}^{\nu} \hat{p}^{\neq j}\, \hat{q}_j\, \hat{s}_k[\eta\eta^{\intercal}].
$$

Equivalently, applying Remark 6 with $k = 2$ and $W_{\neq i_1,i_2} = X \setminus \{x_{i_1}, x_{i_2}\}$, term $B$

follows to be

$$
\begin{aligned}
B \equiv \frac{1}{k!} \iint \sum_{i_1=1}^{k} \sum_{\substack{i_2=1 \\ i_1 \neq i_2}}^{k} \eta(x_{i_1}) \eta(x_{i_2})^{\intercal} \\
\sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} \hat{q}_{j_1} \, \hat{s}_{j_1}(x_{i_1}) \, \hat{q}_{j_2} \, \hat{s}_{j_2}(x_{i_2}) \int\limits_{|W|=k-2} \cdots \int \hat{f}^{\neq j_1, j_2}\left(W_{\neq i_1, i_2}\right) \delta W_{\neq i_1, i_2} \, \mathrm{d}x_{i_1} \mathrm{d}x_{i_2} \\
= \frac{1}{k!} \sum_{i_1=1}^{k} \sum_{\substack{i_2=1 \\ i_1 \neq i_2}}^{k} \sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} \\
\iint \eta(x_{i_1}) \eta(x_{i_2})^{\intercal} \, \hat{q}_{j_1} \, \hat{q}_{j_2} \, \hat{s}_{j_1}(x_{i_1}) \, \hat{s}_{j_2}(x_{i_2}) \, (k-2)! \, \hat{p}^{\neq i_1, i_2}(k-2) \, \mathrm{d}x_{i_1} \mathrm{d}x_{i_2} \\
= \frac{(k-2)!}{k!} \sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} \hat{p}^{\neq j_1, j_2}(k-2) \\
\sum_{i_1=1}^{k} \sum_{\substack{i_2=1 \\ i_1 \neq i_2}}^{k} \hat{q}_{j_1} \, \hat{q}_{j_2} \int \eta(x_{i_1}) \hat{s}_{j_1}(x_{i_1}) \, \mathrm{d}x_{i_1} \int \eta(x_{i_2})^{\intercal} \hat{s}_{j_2}(x_{i_2}) \, \mathrm{d}x_{i_2} \\
= \frac{(k-2)!}{k!} \sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} \hat{p}^{\neq j_1, j_2}(k-2) \sum_{i_1=1}^{k} \sum_{\substack{i_2=1 \\ i_1 \neq i_2}}^{k} \hat{q}_{j_1} \, \hat{q}_{j_2} \, \hat{s}_{j_1}[\eta] \, \hat{s}_{j_2}[\eta] \\
= \frac{k \, (k-1) \, \cancel{(k-2)!}}{k!} \sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} \hat{p}^{\neq j_1, j_2}(k-2) \, \hat{q}_{j_1} \, \hat{q}_{j_2} \, \hat{s}_{j_1}[\eta] \, \hat{s}_{j_2}[\eta]^{\intercal} \\
= \sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} \hat{p}^{\neq j_1, j_2}(k-2) \, \hat{q}_{j_1} \, \hat{q}_{j_2} \, \hat{s}_{j_1}[\eta] \, \hat{s}_{j_2}[\eta]^{\intercal}.
\end{aligned}
$$

Consequently, the second raw moment becomes

$$
\begin{aligned}
\mathbb{E}_{\zeta_{k,1:\nu}}[zz^{\intercal}] = \frac{1}{\hat{p}(k)} \sum_{k=1}^{\nu} \hat{p}^{\neq j}(k-1) \, \hat{q}_j \, \hat{s}_j[\eta\eta^{\intercal}] \\
+ \frac{1}{\hat{p}(k)} \sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} p^{\neq j_1, j_2}(k-2) \, \hat{q}_{j_1} \, \hat{q}_{j_2} \, \hat{s}_{j_1}[\eta] \, \hat{s}_{j_2}[\eta]^{\intercal}
\end{aligned}
$$

and thus the covariance $P_{\zeta_{k,1:\nu}}$ becomes

$$\begin{aligned}
P_{\zeta_{k,1:\nu}} &= \mathbb{E}\Big[(z - m_{\zeta_{k,1:\nu}})(z - m_{\zeta_{k,1:\nu}})^\intercal\Big] \\
&= \mathbb{E}[zz^\intercal] - m_{\zeta_{k,1:\nu}} m_{\zeta_{k,1:\nu}}^\intercal \\
&= \frac{1}{\hat{p}(k)} \sum_{k=1}^{\nu} \hat{p}^{\neq j}(k-1)\, \hat{q}_j\, \hat{s}_j[\eta\eta^\intercal] \\
&\quad + \frac{1}{\hat{p}(k)} \sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} p^{\neq j_1, j_2}(k-2)\, \hat{q}_{j_1}\, \hat{q}_{j_2}\, \hat{s}_{j_1}[\eta]\, \hat{s}_{j_2}[\eta]^\intercal - m_{\zeta_{k,1:\nu}} m_{\zeta_{k,1:\nu}}^\intercal.
\end{aligned}$$

Finally, applying Remark 5 with $k = 1$ and $k = 2$, it follows as proposed

$$\begin{aligned}
P_{\zeta_{k,1:\nu}} = &\frac{k \sum_{k=1}^{\nu} \hat{p}^{\neq j}(k-1)\, \hat{q}_j\, \hat{s}_j[\eta\eta^\intercal]}{\sum_{k=1}^{\nu} \hat{p}^{\neq j}(k-1)\, \hat{q}_j} \\
&+ \frac{k\,(k-1) \sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} p^{\neq j_1, j_2}(k-2)\, \hat{q}_{j_1}\, \hat{q}_{j_2}\, \hat{s}_{j_1}[\eta]\, \hat{s}_{j_2}[\eta]^\intercal}{\sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} p^{\neq j_1, j_2}(k-2)\, \hat{q}_{j_1}\, \hat{q}_{j_2}} - m_{\zeta_{k,1:\nu}} m_{\zeta_{k,1:\nu}}^\intercal.
\end{aligned}$$

Replacing $\hat{q}_j = q_j\, s_j[p_V]$ and $\hat{s}_j(x) = \frac{p_V\, s_j(x)}{s_j[p_V]}$ gives the proposed result

$$\begin{aligned}
P_{\zeta_{n,1:\nu}} = &\frac{k \sum_{j=1}^{\nu} p^{\neq j}(k-1)\, q_j\, s_j[\eta\eta^\intercal\, p_V]}{\sum_{j=1}^{\nu} p^{\neq j}(k-1)\, q_j\, s_j[p_V]} \\
&+ \frac{k\,(k-1) \sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} p^{\neq j_1, j_2}(k-2)\, q_{j_1}\, q_{j_2}\, s_{j_1}[\eta\, p_V]\, s_{j_2}[\eta\, p_V]^\intercal}{\sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} p^{\neq j_1, j_2}(k-2)\, q_{j_1}\, q_{j_2}\, s_{j_1}[p_V]\, s_{j_2}[p_V]} \\
&- m_{\zeta_{k,1:\nu}} m_{\zeta_{k,1:\nu}}^\intercal.
\end{aligned}$$

□

## 8.5. Exemplary comparison of the Pseudo–likelihoods

In the previous sections different approaches to approximate the Pseudo–likelihood $\varphi(z)$ were proposed. In the following, the different approaches are summarized and compared exemplarily to each other, in turn trying to emphasize the difference between them. In this example, a particular set of Bernoulli components $\left\{(q_i, s_i(x))_{i=1}^4\right\}$ where the single–object distributions $s_i(x)$ are one–dimensional Gaussians with $\mathcal{N}_{m_i,P_i}(x)$ are chosen. The component parameters are

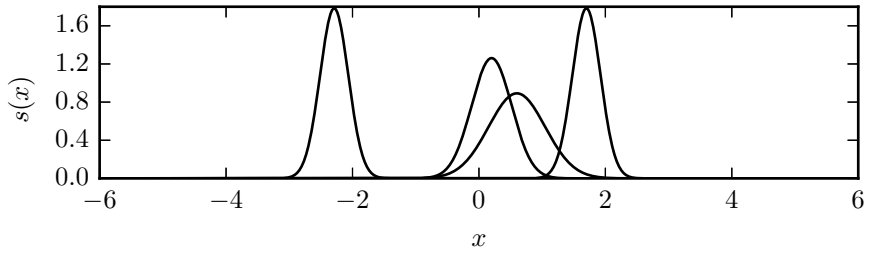


Figure 8.1.: PDFs of the multi–object multi–Bernoulli distribution used for the study. The 4 single–object PDF are plotted separately. The PDFs correspond to the $i$–th Bernoulli component where the left–most corresponds to first component and the right–most to the last component.

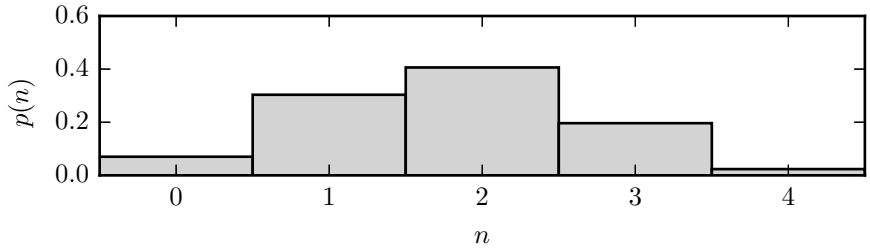


Figure 8.2.: PMF of the multi–object multi–Bernoulli distribution used for the study.

$$q_{1:4} = (0.50, 0.45, 0.70, 0.15),$$
$$m_{1:4} = (-2.3, 0.20, 0.60, 1.70)$$

and

$$P_{1:4} = (0.50, 0.10, 0.20, 0.05).$$

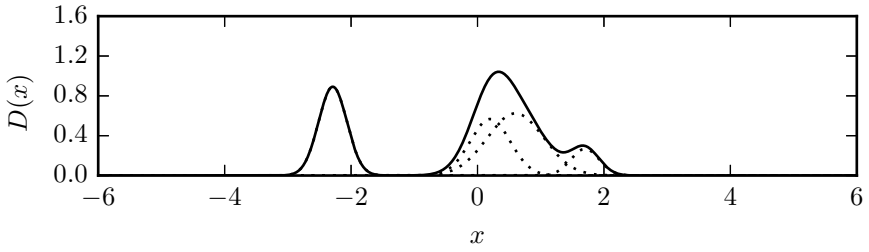


Figure 8.3.: Probability hypothesis density (PHD) of the multi–object multi–Bernoulli distribution used for the study. The single–Bernoulli PHDs (dotted) and the resulting multi–Bernoulli PHD (solid) are shown.

The measurement function in use is linear with $\eta(x) = x$. Using these parameters the PDFs, PMF and PHD are depicted in Figures 8.1, 8.2 and 8.3. Furthermore, given the aforementioned multi–Bernoulli parameters, the resulting approximate Pseudo–likelihoods for the Quasi–Gaussian–Mixture, Quasi–Poisson–Binomial and Quasi–Gaussian as presented in Figure 8.4. It can be recognized that the

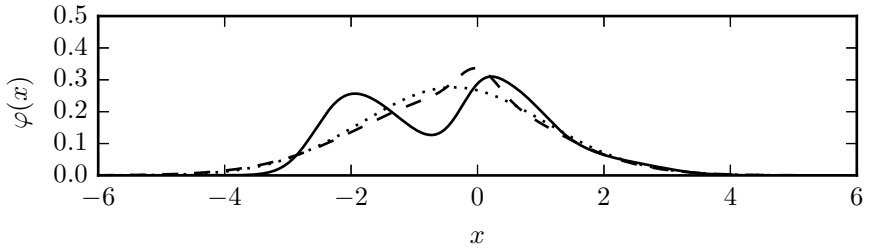


Figure 8.4.: Comparison of the Quasi–Gaussian–Mixture (solid), Quasi–Poisson–Binomial (dashed) and Quasi–Gaussian (dotted) approximations of the Pseudo–likelihood for the multi–Bernoulli distribution used in this study.

different approximations yield different results. Note that the Gaussian–Mixture approximation is equal to the true Pseudo–likelihood in case of this linear Gaussian example. It can be seen that the Gaussian–Mixture approximation has two modes at $x \approx -2.0$ and $x \approx 0.1$, whereas the Poisson–Binomial and the Gaussian approximation have only one mode at $x \approx 0.2$ and $x \approx 0$ respectively. However, there is also a difference between the Poisson–Binomial and the Gaussian recognizable.

# Part IV.

# Implementations of RFS Filter

# 9. Sequential Monte Carlo Implementations

## Contents



## Introduction

In the previous chapters, several variants of superpositional multi–object multi–Bernoulli ($\Sigma$–MeMBer) filters together with computationally tractable approximations were introduced. While those approximations reduce the computational complexity in general, it is still not possible to implement those filters. This is due to the fact that the single–object spatial probability density functions (PDFs) $s_i(x)$ are still arbitrary. In order to make those filters implementable further assumptions and approximations are necessary.

In this section, Sequential Monte Carlo (SMC) implementations of the previously proposed $\Sigma$–MeMBer filter variants are presented. In general, the SMC implementations of the multi–object multi–Bernoulli (MeMBer) filters are a straightforward extension of the well studied Particle filter (PF) used in the single–object Bayes filter case. Thus, given the MeMBer parameter set $\{(q_i, s_i(x))_{i=1}^{\nu}\}$, then the single–object PDFs $s_i(x)$ will be approximated by a set of weighted samples. Therefore, let each single–object PDF $s_i(x)$ be approximated by $J_i$ samples at the

supports $x_{(i,j)}$ having weights $w_{(i,j)}$, then it follows that

$$s_i(x) \approx \sum_{j=1}^{J_i} w_{(i,j)}\, \delta_{x_{(i,j)}}(x).$$

At first, the SMC implementation of the cardinality balanced multi–object multi–Bernoulli (CB-MeMBer) filter predictor and corrector according to [VVC09] are presented as reference. Then, the SMC implementations of the approximate and intensity corrector variant of the $\Sigma$–MeMBer filter are introduced.

## 9.1. SMC CB–MeMBer Filter

In this section, the predictor and corrector of the SMC CB-MeMBer filter are presented.

### 9.1.1. MeMBer Predictor SMC Implementation

Recalling the predictor equations of the MeMBer filter from Proposition 17, then according to [Vo08, Sec. 6.4] the SMC predictor equations are as follows.

Given the single–object Markov transition density $f_{\mathrm{T}}(x \mid x')$ Markov transition, its proposal distribution with PDF $f_p(x \mid x')$, the single–object birth PDF $s_{\mathrm{B}_i}(x)$ and its proposal distribution having PDF $f_{\mathrm{B}_i}(x)$, then the transitioned Bernoulli components become

$$q_{\mathrm{T}_i} = q_i \sum_{j=1}^{J_i} p_S(x_{(i,j)})\, w_{(i,j)}$$

and

$$s_{\mathrm{T}_i}(x) = \sum_{j=1}^{J_i} w_{\mathrm{T}_{(i,j)}}\, \delta_{x_{\mathrm{T}_{(i,j)}}}(x)$$

with elements

$$x_{\mathrm{T}_{(i,j)}} \sim f_p\left(\cdot \mid x_{(i,j)}\right),$$

$$\overline{w}_{\mathrm{T}_{(i,j)}} = \frac{w_{(i,j)}\, p_S\left(x_{(i,j)}\right)\, f_T\left(x_{T_{(i,j)}} \mid x_{(i,j)}\right)}{f_p\left(x_{T_{(i,j)}} \mid x_{(i,j)}\right)}$$

and

$$w_{\mathrm{T}_{(i,j)}} = \frac{\overline{w}_{\mathrm{T}_{(i,j)}}}{\sum_{j=1}^{J_i} \overline{w}_{\mathrm{T}_{(i,j)}}}.$$

Additionally, the new born Bernoulli components are

$$q_{\mathrm{B}_i} \in [0..1]$$

and

$$s_{\mathrm{B}_i}(x) = \sum_{j=1}^{J_i} w_{\mathrm{B}(i,j)}\, \delta_{x_{\mathrm{B}_i(i,j)}}(x)$$

with elements

$$x_{\mathrm{B}_{(i,j)}} \sim f_{\mathrm{B}_i}(\cdot),$$
$$\overline{w}_{\mathrm{B}_{(i,j)}} = \frac{s_{\mathrm{B}_i}\left(x_{\mathrm{B}_{(i,j)}}\right)}{f_{\mathrm{B}_i}\left(x_{\mathrm{B}_{(i,j)}}\right)}$$

and

$$w_{\mathrm{B}_{(i,j)}} = \frac{\overline{w}_{\mathrm{B}_{(i,j)}}}{\sum_{j=1}^{J_i} \overline{w}_{\mathrm{B}_{(i,j)}}}.$$

### 9.1.2. CB–MeMBer Corrector SMC Implementation

Similar results can easily be established for the CB-MeMBer corrector equations. Recalling the corrector equations of the CB-MeMBer filter from Proposition 18, then according to [Vo08, Sec. 6.4] the SMC predictor equations are as follows.

Assuming that $\left\{\left(q_i', s_i'(x)\right)_{i=1}^{\nu'}\right\}$ is the set of predicted Bernoulli parameters with $s_i'(x) = \sum_{j=1}^{J_i} w_{(i,j)}'\, \delta_{x_{(i,j)}'}(x)$, $p_D(x)$ the probability of detection and $f_z(x) \overset{\text{abbr}}{=} f(z \mid x)$ is the single–object single–measurement likelihood, then the SMC approximation of the missed Bernoulli components are

$$q_{m_i} = \frac{1 - \rho_{m_i}}{1 - q_i'\, \rho_{m_i}}\, q_i'$$

and

$$s_{m_i}(x) = \sum_{j=1}^{J_i} w_{m_{(i,j)}}\, \delta_{x_{(i,j)}'}(x)$$

with

$$\rho_{m_i} = \sum_{j=1}^{J_i} w_{(i,j)}'\, p_D(x_{(i,j)}'),$$
$$\overline{w}_{m_{(i,j)}} = \left(1 - p_D(x_{(i,j)}')\right)\, w_{(i,j)}'$$

and

$$w_{m_{(i,j)}} = \frac{\overline{w}_{m_{(i,j)}}}{\sum_{j=1}^{J_i} \overline{w}_{m_{(i,j)}}}.$$

Furthermore, the detected Bernoulli components are

$$q_d(z) \approx \frac{\sum_{j=1}^{\nu'} \frac{(1'-q_j)}{\left(1-q_j'\rho_{m_j}\right)} \frac{q_j'\rho_{d_j}(z)}{\left(1-q_j'\rho_{m_j}\right)}}{\lambda_C f_C(z) + \sum_{j=1}^{\nu'} \frac{q_j'\rho_{d_j}(z)}{1-q_j'\rho_{m_j}}}$$

and

$$s_d(x \mid z) = \sum_{i=1}^{\nu'} \sum_{j=1}^{J_i} w_{d_{(i,j)}}(z)\, \delta_{x'_{(i,j)}}(x)$$

with

$$\rho_{d_i}(z) = \sum_{j=1}^{J_i} p_D\left(x'_{(i,j)}\right)\, w'_{(i,j)}\, f_z\left(x'_{(i,j)}\right),$$

$$\overline{w}_{d_{(i,j)}}(z) = \sum_{j=1}^{\nu'} \frac{q_j'}{1-q_j'}\, w'_{(i,j)}\, f_z\left(x'_{(i,j)}\right)$$

and

$$w_{d_{(i,j)}}(z) = \frac{\overline{w}_{d_{(i,j)}}(z)}{\sum_{i=1}^{\nu'} \sum_{j=1}^{J_i} \overline{w}_{d_{(i,j)}}(z)}.$$

## 9.2. SMC $\Sigma$–MeMBer Filters

In this section, the SMC implementation of the $\Sigma$–MeMBer filter variants from Section 7.2 are presented. The approximation principle is the same as used for the SMC MeMBer filter and as such its derivation is straightforward. Therefore, only the main results are presented. As before, the focus is on the corrector equations only. Since all variants share the same Pseudo–likelihood, the SMC implementations of the Pseudo–likelihoods variants as stated in Chapter 8 are presented in its own paragraph.

### 9.2.1. Approximate Σ–MeMBer Corrector SMC Implementation

Considering the Approximate Σ–MeMBer corrector equations as presented in Theorem 3, then its SMC implementations becomes as stated next.

Given a set of predicted Bernoulli components $\left(\{\left(q_i', s_i'(x)\right)\}\right)$, the Pseudo–likelihood $\varphi^{\neq i}(z)$ as stated in Section 9.3, then the missed Bernoulli components are

$$q_{m_i} = \frac{1-\rho_{m_i}}{1-q_i'\,\rho_{m_i}}\,q_i'$$

and

$$s_{m_i}(x) = \sum_{j=1}^{J_i} w_{m_{(i,j)}}\,\delta_{x'_{(i,j)}}(x)$$

with

$$\rho_{m_i} = \sum_{j=1}^{J_i} w'_{(i,j)}\,p_V(x'_{(i,j)}),$$
$$\overline{w}_{m_{(i,j)}} = \left(1-p_V(x'_{(i,j)})\right)\,w'_{(i,j)}$$

and

$$w_{m_{(i,j)}} = \frac{\overline{w}_{m_{(i,j)}}}{\sum\limits_{j=1}^{J_i}\overline{w}_{m_{(i,j)}}}$$

Moreover, the detected Bernoulli components are

$$q_d(z) = \frac{\frac{(1'-q_i)}{\left(1-q_i'\rho_{m_i}\right)}\,\frac{q_i'\rho_i(z_k)}{\left(1-q_i'\rho_{m_i}\right)}}{\varphi^{\neq i}(z_k)+\frac{q_i'\rho_i(z_k)}{1-q_i'\rho_{m_i}}}$$

and

$$s_{d_i}(x\mid z) = \sum_{j=1}^{J_i} w_{d_{(i,j)}}(z)\,\delta_{x'_{(i,j)}}(x)$$

where

$$\rho_{d_i}(z) = \sum_{j=1}^{J_i} p_V\left(x'_{(i,j)}\right)\,w'_{(i,j)}\,\varphi^{\neq i}\left(z-\eta(x'_{(i,j)})\right),$$
$$\overline{w}_{d_{(i,j)}}(z) = p_V\left(x'_{(i,j)}\right)\,w'_{(i,j)}\,\varphi^{\neq i}\left(z-\eta(x'_{(i,j)})\right)$$

and

$$w_{d_{(i,j)}}(z) = \frac{\overline{w}_{d_{(i,j)}}(z)}{\sum_{j=1}^{J_i} \overline{w}_{d_{(i,j)}}(z)}.$$

### 9.2.2. Intensity $\Sigma$–MeMBer Corrector SMC Implementation

Recalling the Intensity $\Sigma$–MeMBer corrector equations as presented in Theorem 4, then the SMC implementation of the Intensity $\Sigma$–MeMBer corrector equations can be established as stated subsequently.

Note that in contrast to the approximate $\Sigma$–MeMBer corrector, the intensity variant does not make a distinction between missed and detected objects.

Given a set of predicted Bernoulli components $\left(\left\{\left(q_i', s_i'(x)\right)\right\}\right)$, the Pseudo–likelihood $\varphi^{\neq i}(z)$ as stated in Section 9.3, then the corrected Bernoulli components are

$$q_i(z_k) = \frac{\left(q_i' - q_i' \, \rho_{m_i}\right) \varphi^{\neq j}(z_k) + q_i' \, \rho_{d_i}(z_k)}{\left(1 - q_i' \, \rho_{m_i}\right) \varphi^{\neq j}(z_k) + q_i' \, \rho_{d_i}(z_k)}$$

and

$$s_{d_i}(x \mid z_k) = \sum_{j=1}^{J_i} w_{d_{(i,j)}}(z) \, \delta_{x'_{(i,j)}}(x)$$

where

$$\rho_{m_i} = \sum_{j=1}^{J_i} w'_{(i,j)} \, p_V(x'_{(i,j)}),$$

$$\rho_{d_i}(z) = \sum_{j=1}^{J_i} p_V\left(x'_{(i,j)}\right) \, w'_{(i,j)} \, \varphi^{\neq i}\left(z - \eta(x'_{(i,j)})\right),$$

$$\overline{w}_{d_{(i,j)}}(z) = \left(1 - p_V\left(x'_{(i,j)}\right)\right) \, w'_{(i,j)} \, \varphi^{\neq i}(z) + p_V\left(x'_{(i,j)}\right) \, w'_{(i,j)} \, \varphi^{\neq i}\left(z - \eta(x'_{(i,j)})\right)$$

and

$$w_{d_{(i,j)}}(z) = \frac{\overline{w}_{d_{(i,j)}}(z)}{\sum_{j=1}^{J_i} \overline{w}_{d_{(i,j)}}(z)}.$$

## 9.3. Pseudo–likelihoods SMC Implementations

As mentioned previously, the approximate and intensity variants of the $\Sigma$–MeMBer filter share the same Pseudo–likelihood. As described in Chapter 8, there exists different choices for approximating the pseudo–likelihood:

1. A Quasi–Gaussian variant as described in Section .8.2.
2. A Quasi–Gaussian Mixture variant as described in Section 8.3.
3. A Quasi–Poisson–Binomial variant as described in Section 8.4.

In the following a SMC approximation for each of those will be presented. Deriving SMC implementations of the Pseudo–likelihood variants turns out to be simple. In general, the computation of the means and variances of the Gaussians changes by strictly replacing each single–object PDF $s_i'(x)$ with its particle approximation $s_i(x) = \sum_{j=1}^{J_i} w_{(i,j)}\, \delta_{x_{(i,j)}}(x)$.

### 9.3.1. Quasi–Gaussian SMC Implementation

Given the Pseudo–likelihood approximation as presented in Theorem 5, then its SMC implementation becomes as follows.

Let $f_w(z)$ be the PDF of the additive noise, $\eta(x)$ be the single–object measurement function and $\{(q_i, s_i(x))_{i=1}^{\nu}\}$ be the set of Bernoulli parameters, then the SMC implementation of the Quasi–Gaussian Pseudo–likelihood becomes

$$\varphi(z) = \left(f_w * \mathcal{N}_{m_\zeta, P_\zeta}\right)(z)$$

with

$$\begin{aligned} m_\zeta &\triangleq \sum_{j=1}^{\nu} m_{\zeta_j}, \\ P_\zeta &\triangleq \sum_{j=1}^{\nu} P_{\zeta_j}, \\ m_{\zeta_i} &= q_i \sum_{j=1}^{J_i} p_V\left(x_{(i,j)}\right)\, w_{(i,j)}\, \eta\left(x_{(i,j)}\right) \end{aligned}$$

and

$$P_{\zeta_i} = q_i \sum_{j=1}^{J_i} p_V\left(x_{(i,j)}\right)\, w_{(i,j)}\, \eta\left(x_{(i,j)}\right)\eta^{\mathsf{T}}\left(x_{(i,j)}\right) - m_{\zeta_i} m_{\zeta_i}^{\mathsf{T}}.$$

### 9.3.2. Quasi–Gaussian Mixture SMC Implementation

Given the Pseudo–likelihood approximation as presented in Theorem 6, then the SMC implementation of the Quasi–Gaussian Mixture $\Sigma$–MeMBer corrector equations can easily be established.

Let $f_w(z)$ be the PDF of the additive noise, $\eta(x)$ be the single–object measurement function and $\{(q_i, s_i(x))_{i=1}^{\nu}\}$ be the set of Bernoulli parameters, then the SMC implementation of the Quasi–Gaussian–Mixture Pseudo–likelihood becomes

$$\varphi(z) = \sum_{n=0}^{\nu} \sum_{1 \leq i_1 < \ldots < i_n \leq \nu} w_{\hat{s}_{i_{1:n}}} \left( f_w * \mathcal{N} m_{\hat{s}_{i_{1:n}}}, P_{\hat{s}_{i_{1:n}}} \right)(z)$$

with

$$w_{\hat{s}_{i_{1:n}}} = \prod_{i=1}^{\nu} (1 - q_i\ \varrho_i) \left( \prod_{j=1}^{n} \frac{q_{i_j}\ \varrho_i}{1 - q_{i_j}\ \varrho_i} \right),$$

$$m_{\hat{s}_{i_{1:n}}} = \sum_{j=1}^{n} m_{\hat{s}_{i_j}}$$

and

$$P_{\hat{s}_{i_{1:n}}} = \sum_{j=1}^{n} P_{\hat{s}_{i_j}}$$

where

$$\varrho_i = \sum_{j=1}^{J_i} p_V(x_{(i,j)})\ w_{(i,j)},$$

$$m_{\hat{s}_i} = \frac{\sum_{j=1}^{J_i} p_V(x_{(i,j)})\ w_{(i,j)}\ \eta(x_{(i,j)})}{\varrho_i}$$

and

$$P_{\hat{s}_i} = \frac{\sum_{j=1}^{J_i} p_V(x_{(i,j)})\ w_{(i,j)}\ \eta(x_{(i,j)}) \eta^{\mathsf{T}}(x_{(i,j)})}{\varrho_i} - m_{\hat{s}_i} m_{\hat{s}_i}^{\mathsf{T}}.$$

### 9.3.3. Quasi–Poisson Binomial SMC Implementation

Given the Pseudo–likelihood approximation as presented in Theorem 7, then the SMC implementation of the Quasi–Poisson Binomial $\Sigma$–MeMBer corrector equations can easily be established.

Let $f_w(z)$ be the PDF of the additive noise, $\eta(x)$ be the single–object measurement function and $\{(q_i, s_i(x))_{i=1}^{\nu}\}$ be the set of Bernoulli parameters, then the SMC implementation of the Quasi–Poisson Pseudo–likelihood becomes

$$\varphi(z) = \sum_{k=0}^{\nu} p(k) \left( f_w * \mathcal{N}_{m_{\zeta_k}, P_{\zeta_k}} \right)(z)$$

with

$$p(k) = \prod_{i=1}^{\nu} (1 - q_i \, \varrho_i) \sum_{1 \le i_1 < \ldots < i_k \le \nu} \prod_{j=1}^{n} \frac{q_{i_j} \, \varrho_i}{1 - q_{i_j} \, \varrho_i},$$

$$m_{\zeta_{k,1:\nu}} = k \, \frac{\sum_{j=1}^{\nu} p^{\neq j}(k-1) \, q_j \, m_j}{\sum_{j=1}^{\nu} p^{\neq j}(k-1) \, q_j \, \varrho_j}$$

and

$$P_{\zeta_{k,1:\nu}} = \frac{k \sum_{j=1}^{\nu} p^{\neq j}(k-1) \, q_j \, M_j}{\sum_{j=1}^{\nu} p^{\neq j}(k-1) \, q_j \, \varrho_j} + \frac{k \, (k-1) \sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} p^{\neq j_1, j_2}(k-2) \, q_{j_1} \, q_{j_2} \, m_{j_1} \, m_{j_2}^{\mathsf{T}}}{\sum_{j_1=1}^{\nu} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{\nu} p^{\neq j_1, j_2}(k-2) \, q_{j_1} \, q_{j_2} \, \varrho_{j_1} \, \varrho_{j_2}} - m_{\zeta_{k,1:\nu}} m_{\zeta_{k,1:\nu}}^{\mathsf{T}}$$

where

$$\varrho_i = \sum_{j=1}^{J_i} p_V(x_{(i,j)}) \, w_{(i,j)},$$

$$m_i = \sum_{j=1}^{J_i} p_V(x_{(i,j)}) \, w_{(i,j)} \, \eta(x_{(i,j)})$$

and

$$M_i = \sum_{j=1}^{J_i} p_V(x_{(i,j)}) \, w_{(i,j)} \, \eta(x_{(i,j)}) \eta^{\mathsf{T}}(x_{(i,j)}).$$

## 9.4. Resampling

Similar to the single–object particle filter the degeneration of the samples is inevitable. In order to to circumvent this particular problem, it is necessary to perform resampling. Resampling basically means that a new set of equally weighted

particles is generated from the current set of not equally weighted particles. In general each sample is duplicated several times proportional to its initial weight. Thus basically removing particles with low weight and and multiplying samples with high weight.

In general, different approaches to resampling have been proposed in literature. The most prominent ones are

- multinomial resampling,
- residual resampling,
- stratified resampling
- and systematic resampling.

While the basic idea is always similar, all have their advantages and disadvantages ([DC05]). In the multi–object case, resampling is applied to each Bernoulli component individually. In general, the number of samples per component could be different and depending on different parameters like the probability of existence $q_i$ or any other parameter. However, since resampling is not the topic of this study, the number of drawn samples is chosen to be a fixed number $J_i$ for each Bernoulli component. Also, if not stated differently, systematic resampling will be used throughout the numerical studies.

## 9.5. State Estimation and Extraction

The proposed CB-MeMBer and $\Sigma$–MeMBer filters provide us with a certain set of multi–Bernoulli parameters $\{(q_i, s_i(x))_{i=1}^{\nu}\}$ after each timestep. Most of the time, the number of components $\nu$ will not reflect the number of objects $N$ in the monitored area. However, in order to estimate the expected number of objects $N$, several ideas are possible.

1. The number of objects can be determined by choosing all Bernoulli components over a certain threshold $t$, such that

$$N = |\{q_i : q_i > t\}|.$$

2. Also, the number of objects can be determined directly from the mean cardinality of a multi–Bernoulli

$$N = \sum_{i=1}^{\nu} q_i$$

as described in Section 3.4.

3. However, it is also possible to determine the Maximum A Posteriori (MAP) estimate from the probability mass function (PMF)

$$N = \underset{n}{\operatorname{argmax}}\, p(n),$$

whereas the PMF $p(n)$ can be directly determined from the probabilities of existence as described in Section 3.4.

In general, all three choices are possible. Nevertheless, in the remainder of the thesis the number of objects is determined by the third variant as the MAP estimate is in most cases the preferable one.

After determining the number of objects it is still necessary to extract the estimated state from the individual components. In the MeMBer case this turns out to be rather easy, as simply the mean or the mode of the $N$ components with the largest weights can be used. Consequently no clustering or any other computationally demanding calculation is required. Comparing this for example to an independent and identically distributed cluster (IIDC) realization as used in the cardinalized probability hypothesis density (CPHD), this is a huge advantage. As in the IIDC realization, by definition all objects share *one* distribution and as such the individual states have to be extracted by clustering or other similar methods, which is on the one hand not only computational expensive but also has the potential to introduce errors.

Summing it up, the state extraction process for the MeMBer filter variants is computational efficient and the estimated states of the objects $\hat{X}$ are determined by

$$\hat{X} = \{m_1, \ldots, m_N\}$$

where

$$m_i = \sum_{j=1}^{J} w_{(i,j)}\, x_{(i,j)}$$

and

$$N = \underset{n}{\operatorname{argmax}}\, p(n).$$

# 10. Numerical Studies

## Contents



## Introduction

In this chapter, the performances of the previously proposed Sequential Monte Carlo (SMC) implementations of the superpositional multi–object multi–Bernoulli ($\Sigma$–MeMBer) filters are demonstrated and verified. More specifically, the following filters are compared to each other:

- The cardinality balanced multi–object multi–Bernoulli (CB-MeMBer) filter,
- the Thouin–Nannuru–Coates (TNC) multi–object multi–Bernoulli (MeMBer) filter,
- the Intensity Quasi Gaussian (IQG) $\Sigma$–MeMBer filter,
- the Approximate Quasi Gaussian (AQG) $\Sigma$–MeMBer filter,
- the Intensity Quasi Poisson Binomial (IQPB) $\Sigma$–MeMBer filter,
- the Approximate Quasi Poisson Binomial (AQPB) $\Sigma$–MeMBer filter,
- the Intensity Quasi Gaussian Mixture (IQGM) $\Sigma$–MeMBer filter and
- the Approximate Quasi Gaussian Mixture (AQGM) $\Sigma$–MeMBer filter.

The general goal is to show that the $\Sigma$–MeMBer filters are applicable and provide the promised advantages when tracking multiple objects with superpositional (SPS) sensors. As an example, multiple thermopile arrays are used to track multiple objects in a monitored area using the raw sensor measurements provided by the thermopiles. In the upcoming sections, the setup of the environment, the used motion model, and the thermopile measurement model are

described. Subsequently, the performance of the proposed filters is studied under various circumstances. The evaluations described in the following sections are statistical evaluations. For examples of real tracking results one is referred to B.

## 10.1. State and Motion Model

In order to study the performance of the various proposed $\Sigma$–MeMBer filters the following setup is used. The monitored area is a square two dimensional area of $4m^2$. Different amounts of objects are present in the monitored area at the same time and objects are created at the boundaries of the region but object death can appear at any time and any location. The different tracks traveled by the objects are depicted in Figure 10.1. Each object's state is described by $x = (p_x, p_y, v_x, v_y, r, a)^\intercal$, where $p_x, p_y$ is the object position in the two dimensional Cartesian plane, $v_x, v_y$ are the respective velocities, $r$ is the object radius and $a$ is the signal strength/amplitude emitted by the object. Note that the formulation of the state includes a radius which defines the physical extend of an object which in turn violates one of the major assumptions made during the derivation of the MeMBer filters. Consequently, the MeMBer filters are not build to handle things like shadowing explicitly. However, it will be shown that these effects can be handled by the fact that objects may be not visible or detectable for a certain amount of time.

The transition of each object is linear but subject to random perturbations. Here, the object state after transition $x' = (p'_x, p'_y, v'_x, v'_y, r', a')^\intercal$ is

$$\begin{aligned}
p'_x &= p_x + T\ v_x + {}^{T^2}\!/_2\ u_x \\
p'_y &= p_y + T\ v_y + {}^{T^2}\!/_2\ u_y \\
v'_x &= v_x + T\ u_x \\
v'_y &= v_y + T\ u_y \\
r' &= r + u_r \\
a' &= a + u_a,
\end{aligned}$$

where the input vector $u = (u_x, u_y, u_r, u_a)^\intercal$ is sampled from a Gaussian random vector with zero mean and covariance $P = \mathrm{diag}(\sigma_x^2, \sigma_y^2, \sigma_r^2, \sigma_y^2)$. Furthermore, the probability of survival $p_s$ is assumed to be constant.

The object birth is modeled by a single Bernoulli component with probability of existence $q_\mathrm{B} = 0.03$, and all objects are assumed to be born at the perimeter of the monitored area. Note that the upcoming parameters are chosen to resemble the motion of human like objects in the monitored area. Therefore, the initial velocities $v_x$ and $v_y$ are sampled from a uniform distribution $\mathcal{U}(0, 0.5)$, thus resembling slow human walking speed. Additionally, the amplitude $a$ is sampled from a uniform distribution $\mathcal{U}(0.5, 0.6)$, which roughly corresponds to $\pm 10\%$ of

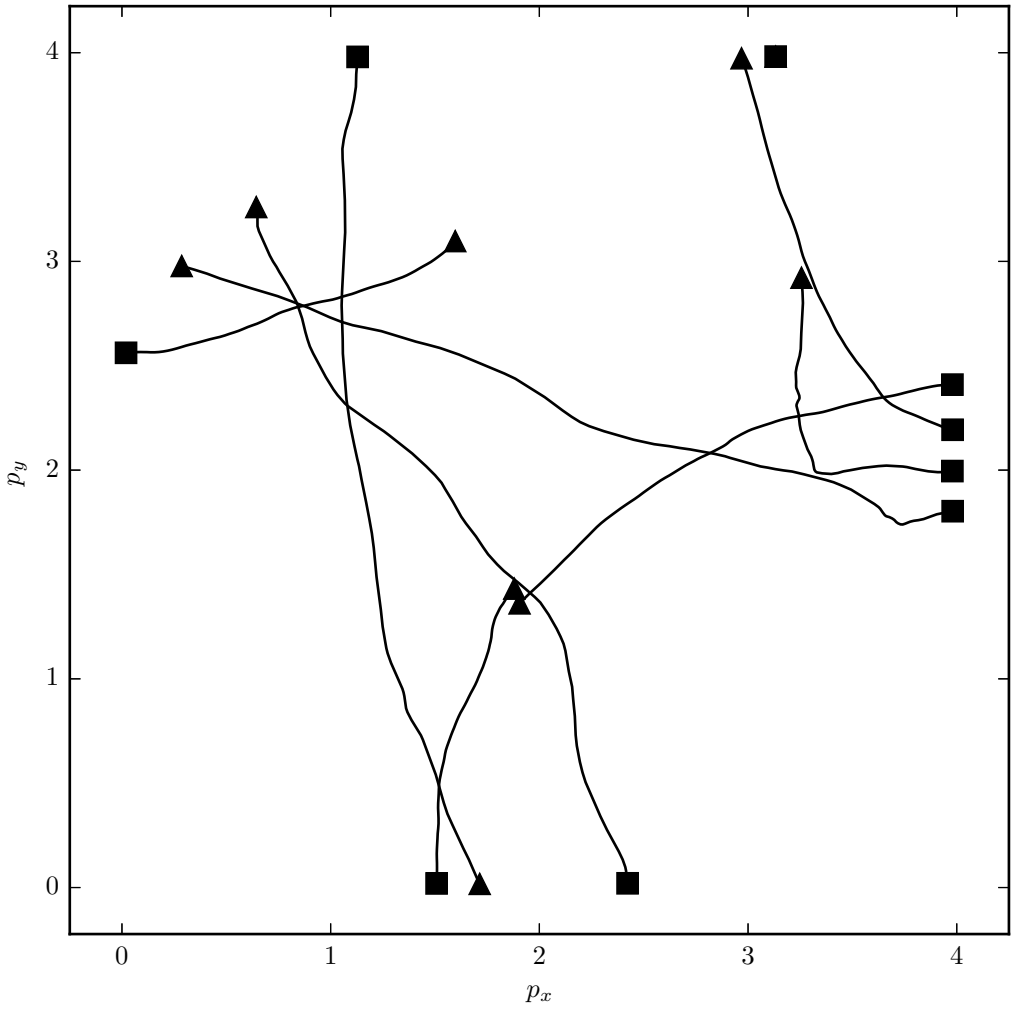


Figure 10.1.: The setup used in the studies. The monitored area is 4 $m^2$ and all objects are born on the perimeter. The squares denote the positions of the object birth and the triangles mark the location of the object death.

Table 10.1.: Parameters used for the transition model.

| T | $\sigma_x$ | $\sigma_y$ | $\sigma_r$ | $\sigma_a$ | $p_s$ | $q_b$ |
|---|---|---|---|---|---|---|
| 0.04 | 0.8 | 0.8 | 0.001 | 0.002 | 0.99 | 0.03 |

Table 10.2.: Lifetime of the individual objects.

| **Object** | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| $\mathbf{T}_{birth}$ | 0 | 13 | 39 | 52 | 65 | 143 | 156 | 169 | 247 |
| $\mathbf{T}_{death}$ | 130 | 163 | 249 | 145 | 249 | 249 | 240 | 249 | 249 |

the irradiance emitted by an object with temperature $T_{obj} = 32°C$, a sensor temperature of $T_{Sensor} = 23°C$ and object emissivity factor 0.98. Here, the sensor temperature is equal to the average room temperature and the emissivity as well as the object temperature is equal to the average emissivity and temperature. Also, the object radius $r$ is sampled from $\mathcal{U}(0.1, 0.25)$ and as such corresponds to the torso size of a human. An overview of the used parameters is shown in table 10.1.

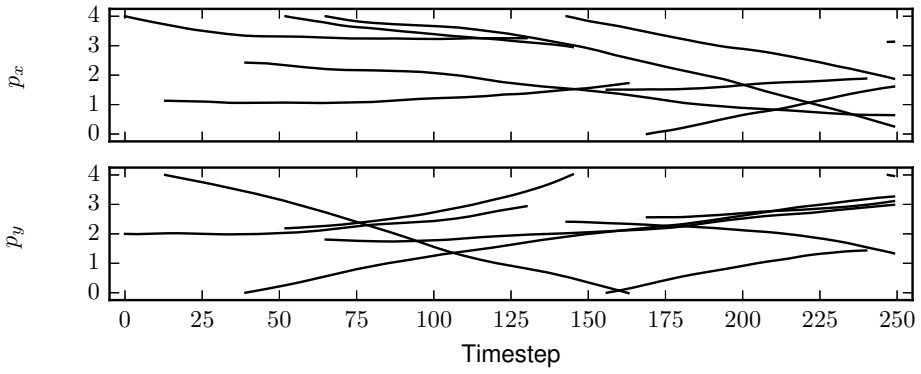


Figure 10.2.: The evolution of the state parameters $p_x$ and $p_y$. Altogether there are 9 different objects entering the monitored area, whereby 6 of them are alive at the same time at most.

The lifetime of the individual objects, including their timestep of birth and death are stated in table 10.2. Furthermore, the resulting tracks and the evolution of the state parameters are shown in Figures 10.2, 10.3 and 10.4.

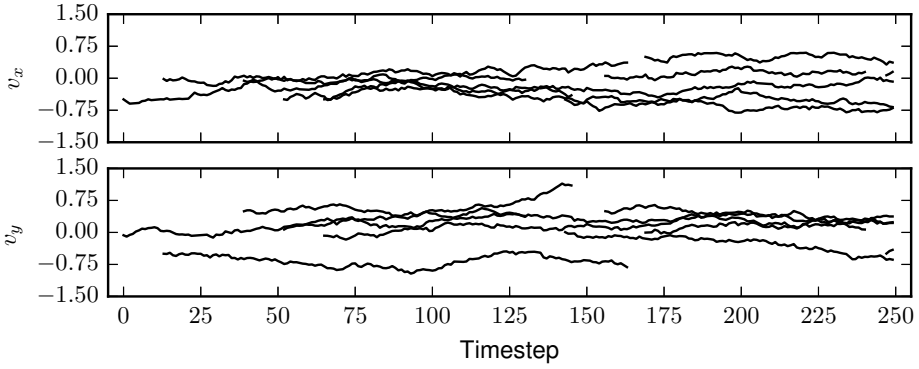


Figure 10.3.: The evolution of the state parameters $v_x$ and $v_y$. Altogether there are 9 different objects entering the monitored area, whereby 6 of them are alive at the same time at most.

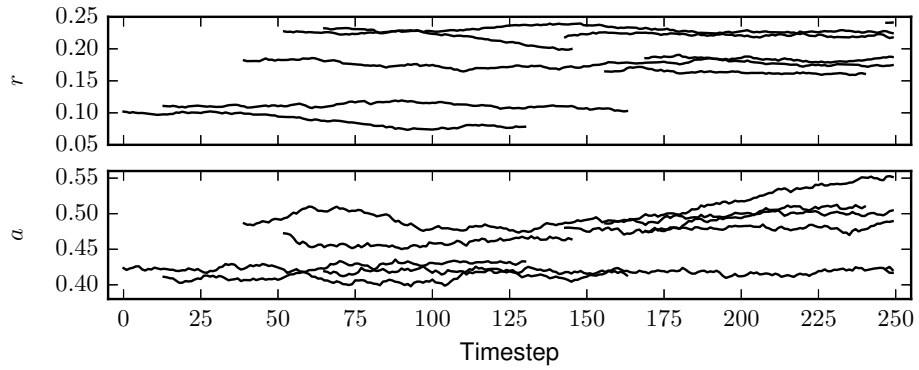


Figure 10.4.: The evolution of the state parameters $r$ and $a$. Altogether there are 9 different objects entering the monitored area, whereby 6 of them are alive at the same time at most.

## 10.2. Measurement Model

In this study, thermopile arrays as described in Section 1.1.1 are used as an example for a SPS–type sensor. For this study, eight thermopile arrays with each eight pixels are placed in pairs in the corners of the monitored area. Each pair of sensors forms a virtual sensor with a nominal field of view (FOV) of approximately ±45° and thus covers the whole monitored area. The measurements of all sensors are collected and treated as single SPS–type measurement with 64 pixels in total.

The non–linear single–pixel measurement function for $n$ objects having state $X = \{x_1, \ldots, x_n\}$ can be described as

$$\eta_j(X) = \sum_{i=1}^{n} \int_{-\Delta_i}^{+\Delta_i} d_j \; a_i \; cos^b(\alpha + \phi_i + \theta_j) \; \mathrm{d}\alpha + \mathbf{w}_j$$

where $\mathbf{w}$ is the additive noise, which is distributed according to a Gaussian distribution with zero mean and variance $\sigma_w^2$. Note that the measurement function uses polar coordinates instead of Cartesian coordinates as employed in the state model. However, the parameters $\phi_i$ and $\Delta_i$ can be deduced from the object Cartesian position $p_x, p_y$ and radius $r_i$ by using simple geometric calculation. For evaluation purposes each single–object probability density function (PDF) is approximated by $J_i = 1000$ particles. Furthermore, the number of kept Bernoulli components is chosen to be twice as much as the maximum number of objects $N_{max} = 6$. Thus, the maximum number of Bernoulli components is $\nu = 2\ N_{max} = 12$. The values of all the parameters are listed in table 10.3.

Table 10.3.: Parameters used for the measurement model.

| $b$ | $d_j$ | $\sigma_w$ | $p_{d/v}$ | $J_i$ | $N_{max}$ | $\nu$ |
|---|---|---|---|---|---|---|
| 280 | 0.006 | 0.10 | 0.95 | 1000 | 6 | 12 |

## 10.3. Performance Metric and Miss Distance

In single–object filtering scenarios, the concept of a miss–distance is taken for granted. However, in multi–object filtering there was for a long time no convention which metric to use. However, in 2010 the Optimal Subpattern Assignment (OSPA) metric was introduced by Ristic, Vo, and Clark. Without going into much detail, the formal definition of the OSPA is as follows. Given two finite sets (FSs) $X = \{x_1, \ldots, x_n\}$ and $Y = \{y_1, \ldots, y_n\}$, cutoff parameters $c > 0$ and order $p > 1$, then the OSPA metric is defined as

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

and

$$d_c(x,y) \triangleq \min(c, d(x,y))$$

where $d(x,y)$ is an arbitrary vector–valued metric and $\Pi_n$ are all possible Permutations of the natural numbers $\{1, \ldots, n\}$. In this study the vector norm $d(x,y) = \|x - y\|$ is used and the parameters are chosen to be $c = 16$ and $p = 1$.

## 10.4. Monte Carlo Verification

The performance of the $\Sigma$–MeMBer filters is evaluated using 100 Monte Carlo (MC) trials. The tracks are always the same but in each trial the measurements are generated randomly. Meaning that the number of generated measurements depends on the probability of detection and the measurements are subject to the random additive noise.

The results for the cardinality estimates are depicted in Figures 10.5, 10.6 and 10.7. Here, the number of objects is estimated using the Maximum A Posteriori (MAP) estimate as described Section 9.5.

For comparison reasons, the results for the SMC CB-MeMBer filter and the TNC–MeMBer filter are also shown. The TNC–MeMBer filter as proposed by Nannuru and Coates in [NC13] can be regarded as specialization of the $\Sigma$–MeMBer filter when the Pseudo–likelihood is chosen to be Quasi–Gaussian and the probability of visibility $p_v(x) = 1$, such that it basically does not account for any missed detections. The CB-MeMBer filter is known to be based on the detection–type model and thus cannot operate on the raw SPS measurement directly. Therefore, separated detection–type measurements with no false measurements are provided to the filter.

Taking a closer look at figure 10.5, it is recognizable that the cardinality estimates of the TNC–MeMBer filter almost always have a large negative bias, which

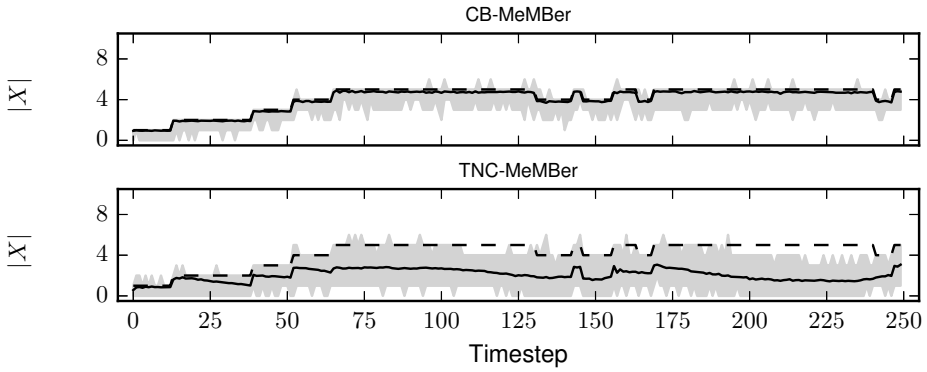


Figure 10.5.: Cardinality estimates over all MC trials of the CB-MeMBer and TNC–MeMBer filters. The graphs show the true cardinality (dotted), the mean cardinality (solid) and standard deviation over all trials (grey area).

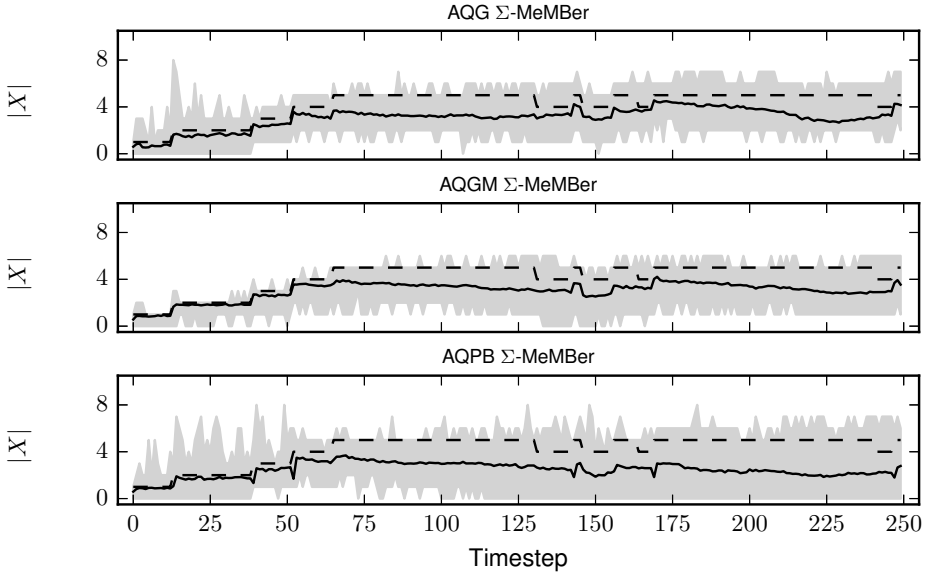


Figure 10.6.: Cardinality estimates over all MC trials of the Approximate Σ–MeMBer filters. The graphs show the true cardinality (dotted), the mean cardinality (solid) and standard deviation over all trials (grey area).

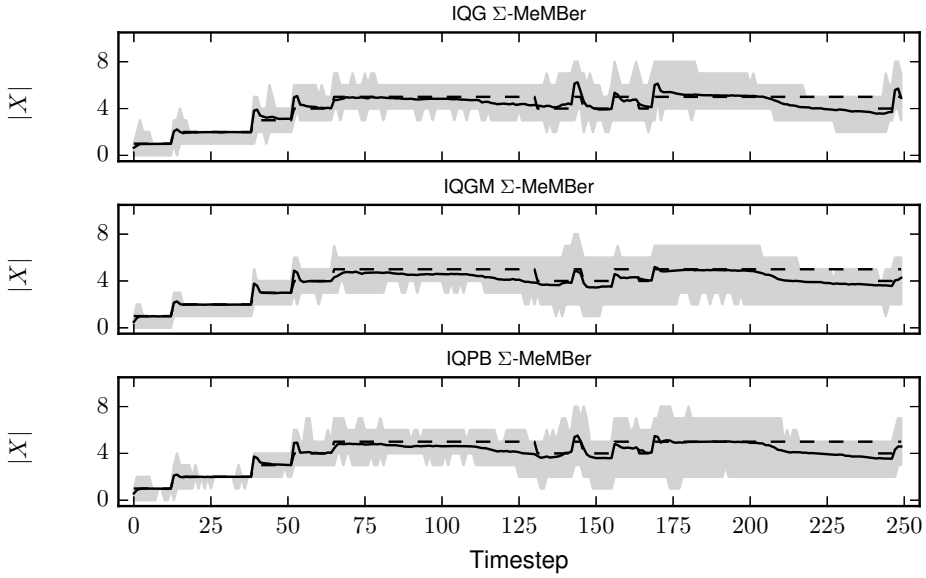


Figure 10.7.: Cardinality estimates over all MC trials of the Intensity $\Sigma$–MeMBer filters. The graphs show the true cardinality (dotted), the mean cardinality (solid) and standard deviation over all trials (grey area).

is more or less what was expected as the TNC filter does not account for missed objects/detections.

In contrast, the results for the $\Sigma$–MeMBer filters can be divided into two camps depending on the type of realization of the filter. One the one side, there are the Approximate $\Sigma$–MeMBer filter variants (see fig. 10.6). While they provide a slightly better cardinality estimate than the TNC–MeMBer filters, they still suffer from huge variants in the cardinality estimates over the various runs. Also, the number of objects is underestimated when the number of objects in the monitored area increases. On the other side, there are the Intensity $\Sigma$–MeMBer filter variants (see fig. 10.7). All three variants provide an accurate and almost similar cardinality estimate and exhibit only major variations over the various MC trials.

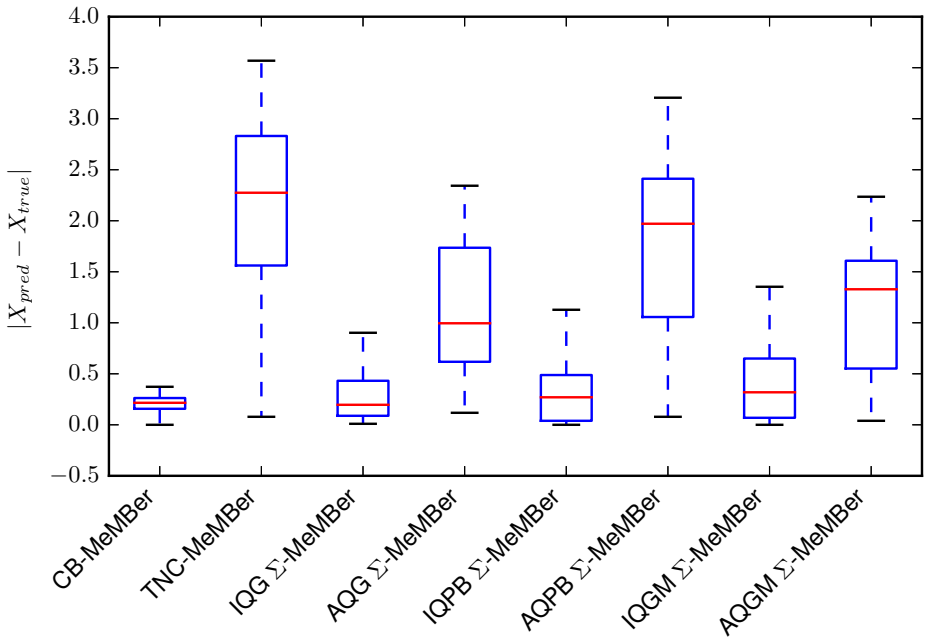


Figure 10.8.: Box–Whisker plot over the cardinality estimates error over all MC trials.

In order to get a better insight about the difference between those filters, consult the Box–Whisker plot of the cardinality error as depicted in Figure 10.8.

The results support the aforementioned insights. It can be seen that the TNC–MeMBer filter provides the worst cardinality estimates followed by the Approximate $\Sigma$–MeMBer filter variants. Clearly, the Intensity $\Sigma$–MeMBer filter provide superior results.

Though, the result for the CB-MeMBer filter seems to be a bit surprising as the CB-MeMBer should outperform the $\Sigma$–MeMBer filters. However, while it does

outperform the Approximate $\Sigma$–MeMBer filters, it does not outperform then Intensity $\Sigma$–MeMBer filters. The reason is that all measurements are subject to the same additive measurement noise. However, in case of the CB-MeMBer filter, the noise is applied to each individual measurement and as such the overall measurement noise in the CB-MeMBer case is higher and consequently filtering becomes harder. Also, the CB-MeMBer filter is known to be overconfident due to the approximations made during the derivation as described in Section 7.1.2.

## 10.5. Miss–Distance Performance

After taking a look at the performance of the filters in terms of estimating the correct number of objects, the focus will now be on the overall miss-distance or OSPA performance. In contrast to the pure cardinality error, it will reflect the overall performance in terms of state and cardinality estimation. The parameters chosen for this study are $p = 1$ and $c = 16$. Therefore, the maximum error penalty $c$ is chosen to be the largest possible distance in the monitored area of 4 $m^2$.

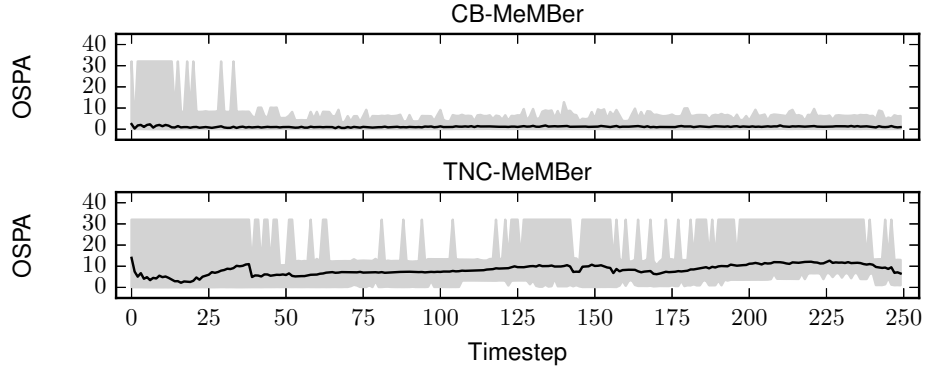


Figure 10.9.: Temporal progress of the OSPA distance with parameters $p = 1$ and $c = 16$ for the CB-MeMBer and TNC–MeMBer filters. The solid line is the average OSPA and the grey area visualizes the minimum and maximum errors over all MC trials.

Having a closer look at the temporal progress of the OSPA distance as depicted in Figures 10.9,

10.10 and 10.11, then the impression of the previous section is confirmed. The Intensity variants of the $\Sigma$–MeMBer filters outperform their Approximate counterparts. Also, the main difference between the different Pseudo–likelihood approximations seems to manifest in the stability of the estimates. The higher the complexity, the more stable the estimates and as such less variations in the OSPA.

For further analyses a Box–Whisker plot of the OSPA distances is shown in Figure 10.12. It can be seen that the median OSPA follows the same pattern as

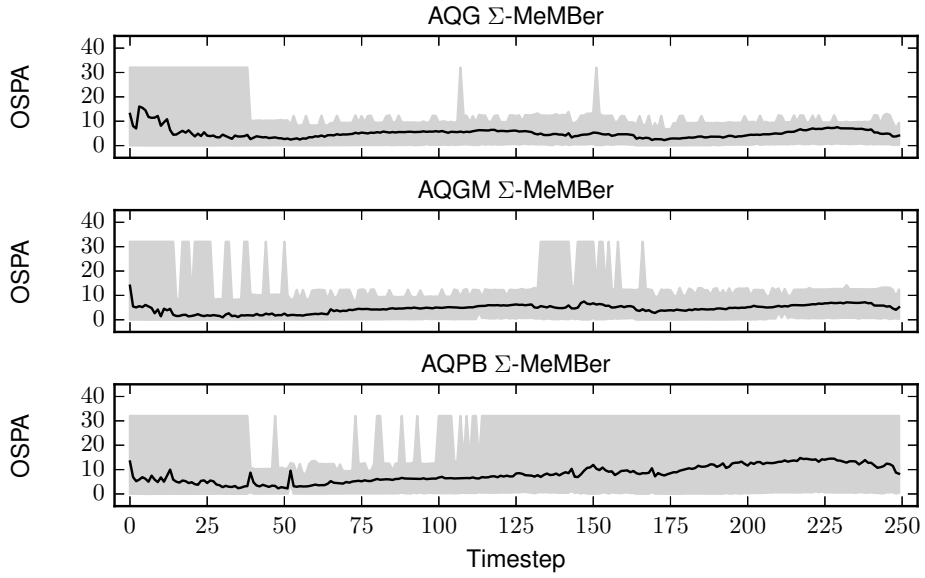


Figure 10.10.: Temporal progress of the OSPA distance with parameters $p = 1$ and $c = 16$ for the Approximate $\Sigma$–MeMBer filters. The solid line is the average OSPA and the grey area visualizes the minimum and maximum errors over all MC trials.

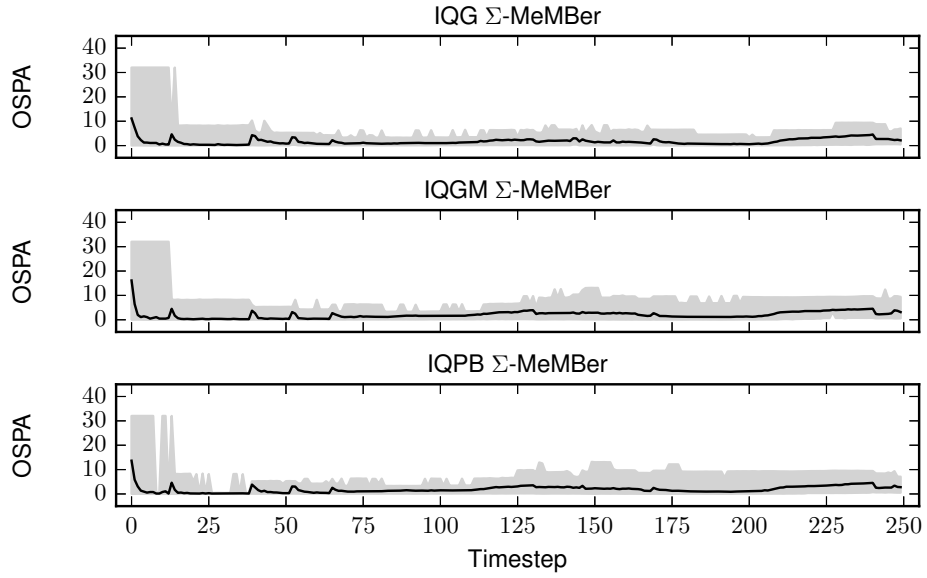


Figure 10.11.: Temporal progress of the OSPA distance with parameters $p = 1$ and $c = 16$ for the Intensity Σ–MeMBer filters. The solid line is the average OSPA and the grey area visualizes the minimum and maximum errors over all MC trials.

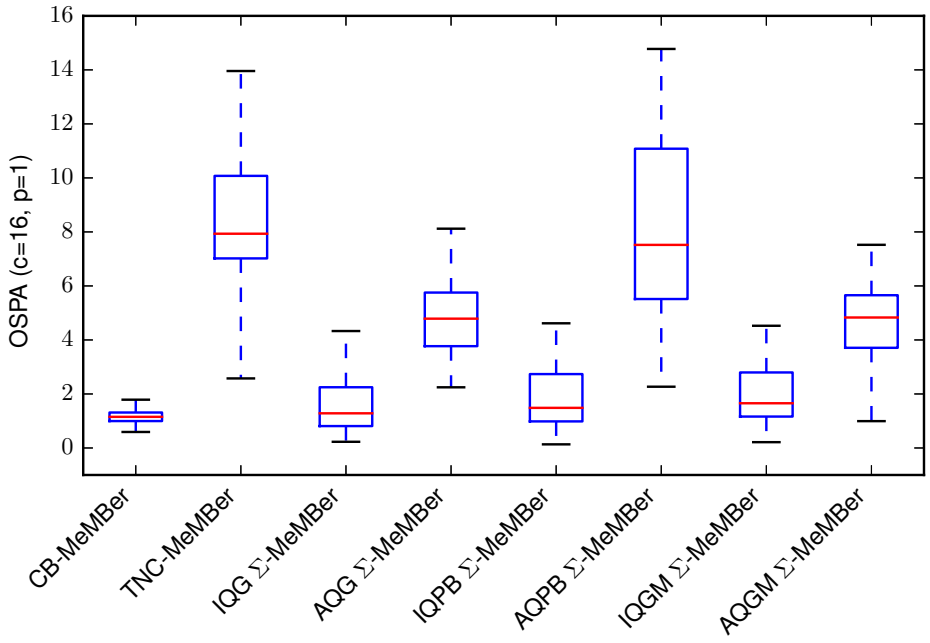


Figure 10.12.: Box–Whisker plot of the OSPA distances with parameters $p = 1$ and $c = 16$ of the $\Sigma$–MeMBer filters, the CB-MeMBer filter and TNC–MeMBer filter.

the OSPA of cardinality estimates. The Intensity $\Sigma$–MeMBer filters outperform their counterparts and the OSPA decreases in terms of its median and variations with increasing filter complexity. Still all $\Sigma$–MeMBer filter variants outperform the $TNC$–MeMBer filter.

## 10.6. Testing of Limitations

After studying the general performance of the proposed $\Sigma$–MeMBer filters, the limitations of the filters are analyzed. In general, there are several parameters that influence the tracking performance of the filters. However, most important seems to be the influence of the amount of missed detections on the tracking performance since their is always the risk that the $\Sigma$–MeMBer filters may loose track of the number of objects and their states completely. In this study, the probability of detection/visibility is varied in the range $p_{D/V} \in [0.5, 1.0]$ with 100 MC runs for each probability of detection. Everything else stays as described in the previous sections.

The results are shown in Figures 10.13 to 10.15. Figure 10.13 shows the OSPA

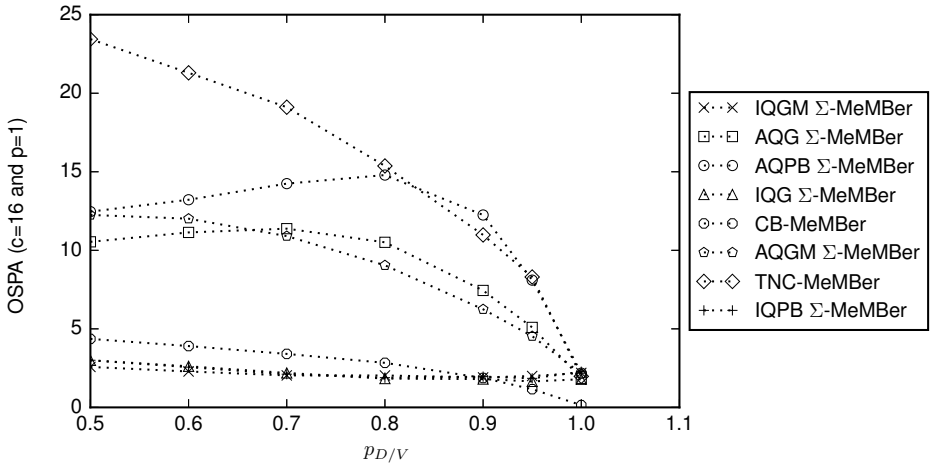


Figure 10.13.: Average OSPA distances for varying probability of visibility. The proposed $\Sigma$–MeMBer filters outperform the TNC–MeMBer filters. However, the Intensity $\Sigma$–MeMBer filters are superior to the Approximate $\Sigma$–MeMBer filters.

with parameters $p = 1$ and $c = 16$. It can be seen that the OSPA decreases for all filters with increases probability of detection/visibility. Again, the Intensity $\Sigma$–MeMBer filters perform best. Since the OSPA does not give a hint if the errors are due to the cardinality or the estimates states, a closer look will be taken on the cardinality and state errors separately. Having a closer look at Figure 10.14

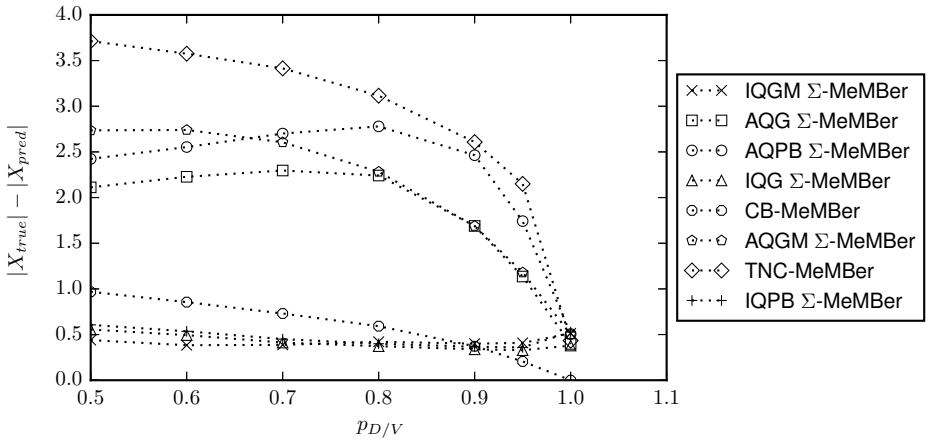


Figure 10.14.: Average cardinality error for varying probability of visibility. Unsurprisingly, the CB-MeMBer filter has the lowest average cardinality error. The $\Sigma$–MeMBer filters perform outstanding even when the probability of detection is very low.

reveals that for example the CB-MeMBer filter has the lowest cardinality error. In turn, meaning that most errors are in the states itself. Furthermore, it can be seen that the Intensity $\Sigma$–MeMBer filters are providing really good cardinality estimates even up to $p_{D/V} = 0.5$ which basically says that in average 50% of the object signals are not seen by the sensor. Finally, Figure 10.15 depicts the OSPA with parameters $p = 1$ and $c = 0$, as such only regarding the errors in the state and not the cardinality.

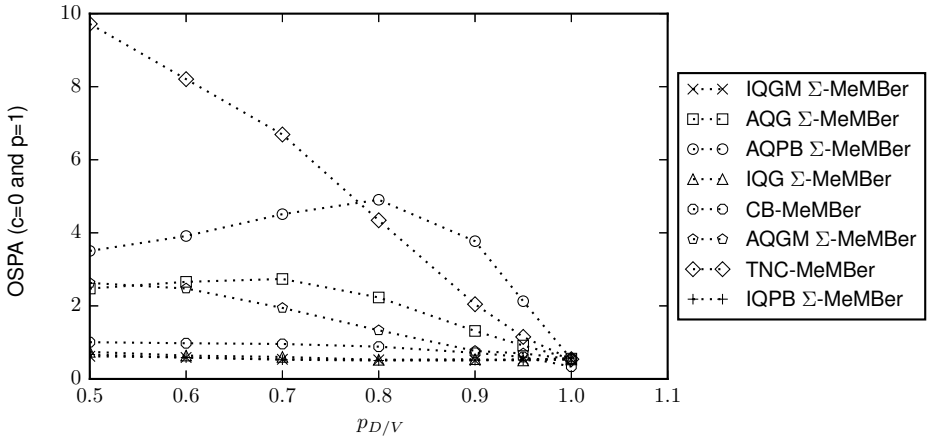


Figure 10.15.: Average OSPA distances for varying probability of visibility with $p = 1$ and $c = 0$. Therefore, only the error in the state estimates is regarded and not the errors in cardinality.

# Part V.

# Final Review

# 11. Summary, Conclusions and Future Research

## Contents



## Introduction

In this chapter the results presented in this thesis are summarized. Hence, the key contributions of this thesis are summarized and conclusions from the results are presented.

## 11.1. Summary

In the beginning of the thesis, several questions arose concerning random finite set (RFS) filters for raw superpositional (SPS) measurements. The first question was, whether it is possible to derive mathematical rigorous multi–object Bayes filters that directly operate on the raw SPS measurement or not.

After an introduction to the thesis in Chapter 1 and to the basics rules of multi–object calculus and finite set statistics (FISST) in Chapter 2, an overview over the most important RFSs was given in Chapter 3.

Hereafter, a measurement model for SPS–type sensors was proposed in Chapter 4, that accounts for the fact that an object might not be visible to a sensor due to being outside of the sensor's restricted field of view (FOV) or because it is occluded by obstacles. Starting from the standard SPS measurement model in Proposition 3, the novel SPS measurement model accounting for missed detections was proposed in Proposition 4.

Then, after explaining multi–object Markov models in Chapter 5, the principal multi–object Bayes filters equations for detection–type and SPS–type sensors were provided in Chapter 6. Hereby, the multi–object Bayes filter corrector equation based on the SPS measurement model was proposed in Proposition 16.

However, the filter equations turned out to be only of theoretical nature and a specific realization for the underlying multi–object distribution had to be chosen. In this theses, the multi–object multi–Bernoulli distribution was chosen and

hence a multi–object multi–Bernoulli (MeMBer) filter for SPS–type sensors was derived in Chapter 7. Nevertheless, while being mathematical rigorous, these filter equations as stated in Theorems 1 and 2 turned out not to be conjugate prior and are as such not recursively applicable over time. That is the reason why, the Approximate superpositional multi–object multi–Bernoulli ($\Sigma$–MeMBer) and the Intensity $\Sigma$–MeMBer filter, were proposed. For the derivation of the Approximate $\Sigma$–MeMBer filter stated in Theorem 1 approximations similar to the ones used by Vo, Vo, and Cantoni in [VVC07] for the derivation of the cardinality balanced multi–object multi–Bernoulli (CB-MeMBer) filter corrector equations were used. For the derivation of the Intensity $\Sigma$–MeMBer filter stated in Theorem 4 the resulting distribution was approximated by the first factorial moment in order to derive valid Bernoulli components.

Starting from these results, computationally tractable variants of the proposed filters were derived in Chapter 8. This included a Gaussian approximation as stated in Theorem 5, a Gaussian Mixture approximation as stated in Theorem 6, and a Poisson Binomial approximation as stated in Theorem 7.

While the proposed filters were in general computationally tractable, more was needed to implement them. Therefore, efficient Sequential Monte Carlo (SMC) implementations for the filters were provided in Chapter 9.

Furthermore, in order to analyze the performance of the filters, numerical experiments were conducted. In these experiments the performances of the proposed filters were measured and compared against the performance of the CB-MeMBer and Thouin–Nannuru–Coates (TNC) filter.

It has been shown that the $\Sigma$–MeMBer filters outperform the TNC filter when the probability that an object is detected by the SPS–type sensor decreases. Furthermore, it was recognizable that the Intensity $\Sigma$–MeMBer filter variants are giving superior results in comparison to the Approximate $\Sigma$–MeMBer filter variants. While the general state estimation seems to be quite similar, the cardinality estimation of the Intensity $\Sigma$–MeMBer filter variants was more accurate and stable. Additionally, it was noticeable that the Intensity $\Sigma$–MeMBer filter variants did not differ much in performance.

## 11.2. Conclusions

Given the results from Chapter 10, some conclusions about the proposed RFS Bayes filters can be drawn. For starters, the $\Sigma$–MeMBer filters provide a promising approach to the problem of multi–object filtering for SPS–type sensors.

Furthermore, the $\Sigma$–MeMBer filter variants offer an improvement over the TNC MeMBer filter variant largely due to the fact that it addresses the issues of missing detection due to occlusions or other effects. Additionally, the Intensity $\Sigma$–MeMBer filter seem to be favorable in comparison to the Approximate $\Sigma$–MeMBer filter variants due to the fact that the cardinality of the latter are far more unstable.

It is also worth mentioning that the filter performance does not differ significantly between the different Pseudo–likelihood approximations. Based on the conducted experiments in this thesis, it seems that the additional computational demand introduced by the Poisson–Binomial and Gaussian Mixture approximations does not improve the filtering results. Therefore, the Gaussian approximation seems to be the mose favorable $\Sigma$–MeMBer filter variant.

## 11.3. Future Research

Future research should analyze how the proposed filters perform in other filtering scenarios. This includes conducting more numerical studies with different motion and measurement models, as well as conducting experiments including data from real sensors. Also, research should be directed at the problem of multi–sensor multi–object filtering, as this is still a partially unsolved problem.

# Part VI.

# Appendix

# A. Key Formulas of Multi–Object Calculus

In this chapter, some key formulas of multi–object calculus are summarized. More information on this topic can be found in [Mah07b, Ch. 11.6].

## Basic rules for Functional Derivatives

### Constant Rule

If the functional $f[h]$ is a constant function $f[h] = K$, then

$$\frac{\delta K}{\delta X} = 0.$$

### Linear Rule

If $f[h] = \int h(x)\, f(x)\, \mathrm{d}x$, then

$$\frac{\delta f[h]}{\delta X} = \begin{cases} f[h] & \text{if } X = \emptyset \\ f(x) & \text{if } X = \{x\} \\ 0 & \text{else.} \end{cases}$$

### Sum Rule

$$\frac{\delta}{\delta X} \sum_{i=1}^{n} F_i[h] = \sum_{i=1}^{n} \frac{\delta F_i[h]}{\delta X}$$

### Product Rule

Given the product of multiple functionals $F_i[h]$, then its derivative with respect to $X$ is

$$\frac{\delta}{\delta X} \prod_{i=1}^{n} F_i[h] = \sum_{\biguplus_{i=1}^{n} W_i = X} \prod_{i=1}^{n} \frac{\delta F_i[h]}{\delta W_i}$$

where the sum is taken over all mutual disjoint subsets $W_1, \ldots, W_n$ of $X$ including the empty set $\emptyset$ such that the union of all subsets equals $\biguplus_{i=1}^{n} W_i = X$. For the special case of $X = \{x\}$ it becomes

$$\frac{\delta}{\delta x} \prod_{i=1}^{n} F_i[h] = \sum_{i=1}^{n} \frac{\delta F_i[h]}{\delta x} \frac{\prod_{j=1}^{n} F_j[h]}{F_i[h]}$$

## Power Rule

The power rule is a specialized variant of the product rule where all functionals are equivalent. Then, it follows

$$\frac{\delta}{\delta x} F[h]^n = n \, F_i[h]^{n-1} \, \frac{\delta F[h]}{\delta x}.$$

## Convolution Rule

The convolution rule follows directly from the product rule. Given the convolution of multiple functionals $F_i[h](z)$, then its derivative with respect to $X$ is

$$\frac{\delta}{\delta X} \left( \overset{n}{\underset{i=1}{\circledast}} F_i[h] \right)(z) = \sum_{\biguplus_{i=1}^{n} W_i = X} \left( \overset{n}{\underset{i=1}{\circledast}} \frac{\delta F_i[h]}{\delta W_i} \right)(z).$$

with

$$\left( \overset{n}{\underset{i=1}{\circledast}} F_i[h] \right)(z) \triangleq (F_1[h] * \ldots * F_n[h])(z).$$

For the special case of $X = \{x\}$ it becomes

$$\frac{\delta}{\delta x} \left( \overset{n}{\underset{i=1}{\circledast}} F_i[h] \right)(z) = \sum_{i=1}^{n} \left( \frac{\delta F_i[h]}{\delta x} * \overset{n}{\underset{\substack{j=1 \\ j \neq i}}{\circledast}} F_j[h] \right)(z)$$

where the convolution is done for all $F_j[h]$ but not the $j$–th.

# B. Examples of estimated Tracks

## Contents



This chapter provides some examples of estimated tracks that have been generated using the exact same parameters as used in Chapter 10.

## B.1. Examples Track with Probability of Detection $p_d = 1.0$

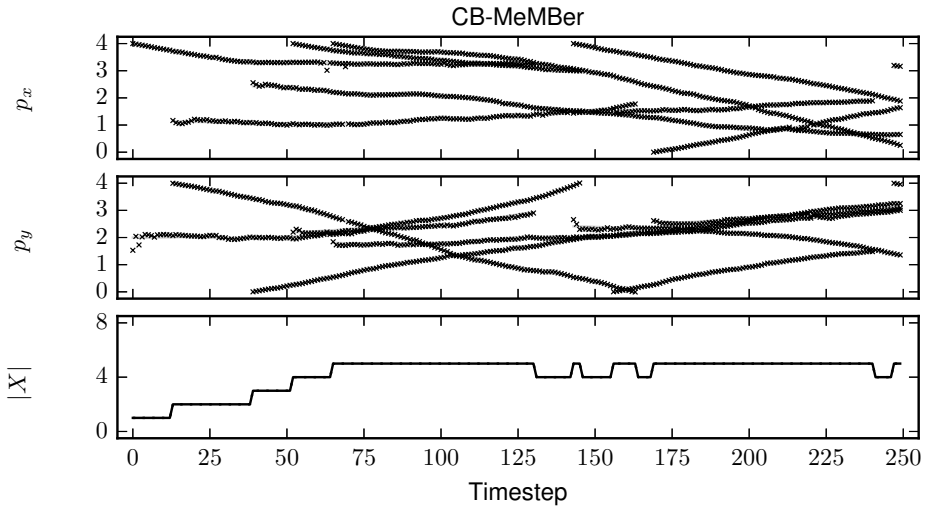


Figure B.1.: Exemplary tracking result for the cardinality balanced multi–object multi–Bernoulli (CB-MeMBer) filter with $p_d = 1.0$. The dotted line is the ground truth and the solid line is the estimated track.

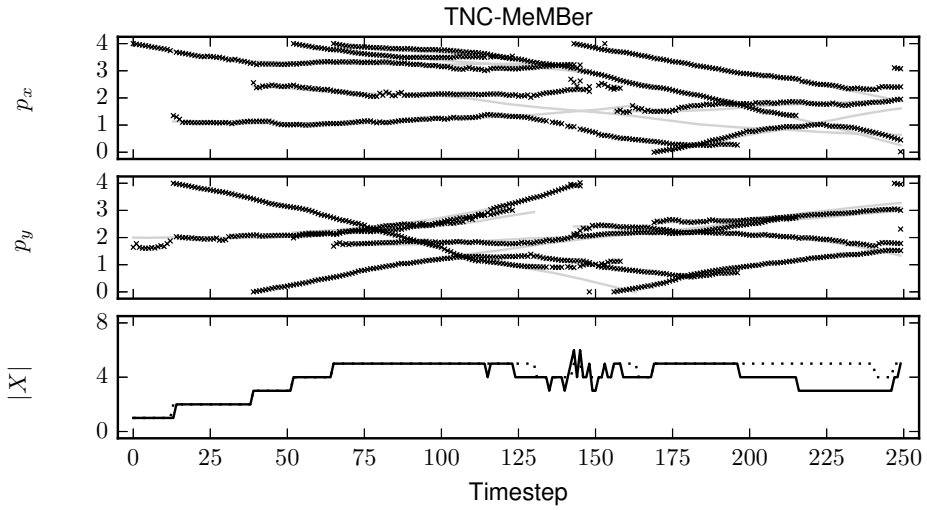


Figure B.2.: Exemplary tracking result for the Thouin–Nannuru–Coates (TNC) multi–object multi–Bernoulli (MeMBer) filter with $p_d = 1.0$. The dotted line is the ground truth and the solid line is the estimated track.

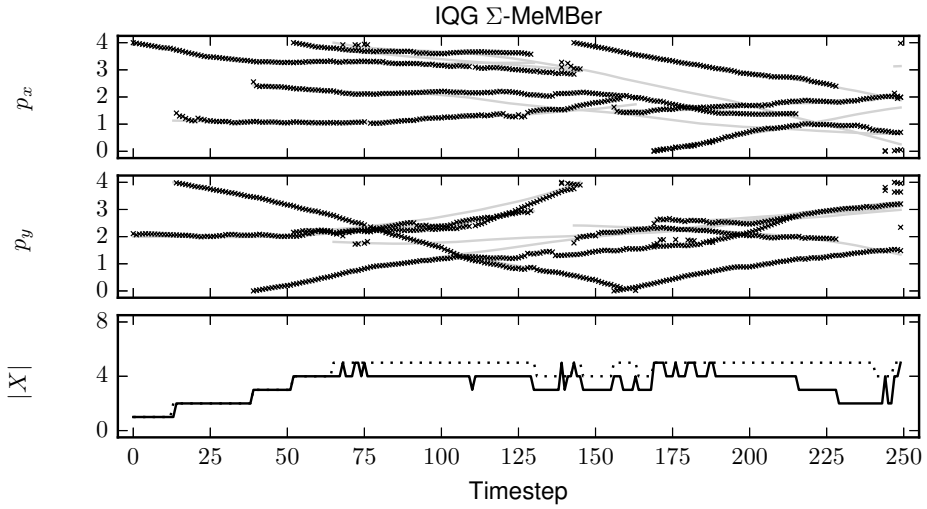


Figure B.3.: Exemplary tracking result for the Intensity Quasi Gaussian (IQG) superpositional multi–object multi–Bernoulli ($\Sigma$–MeMBer) filter with $p_d = 1.0$. The dotted line is the ground truth and the solid line is the estimated track.

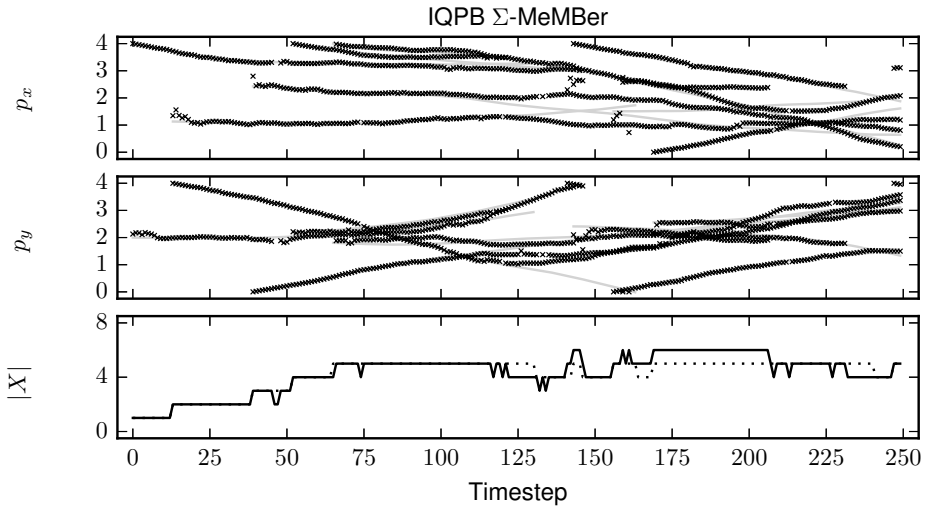


Figure B.4.: Exemplary tracking result for the Intensity Quasi Poisson Binomial (IQPB) CB-MeMBer filter with $p_d = 1.0$. The dotted line is the ground truth and the solid line is the estimated track.

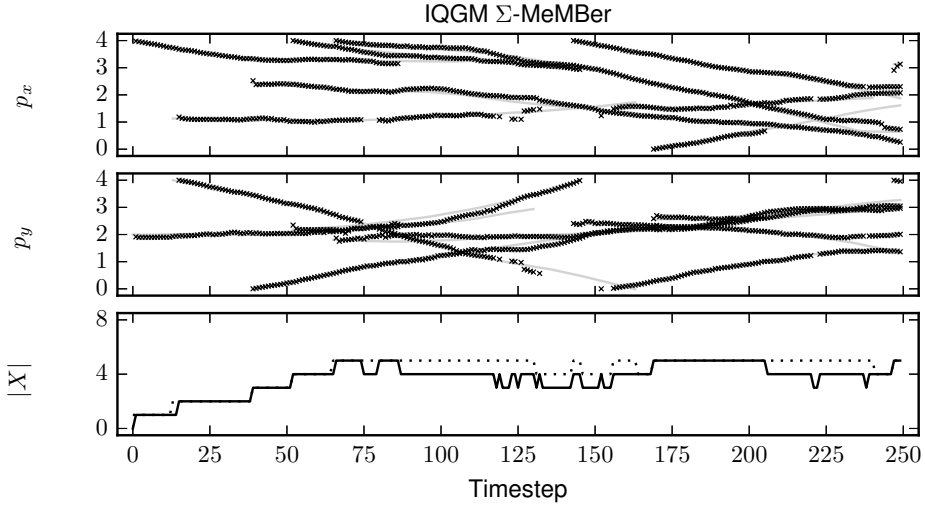


Figure B.5.: Exemplary tracking result for the Intensity Quasi Gaussian Mixture (IQGM) CB-MeMBer filter with $p_d = 1.0$. The dotted line is the ground truth and the solid line is the estimated track.

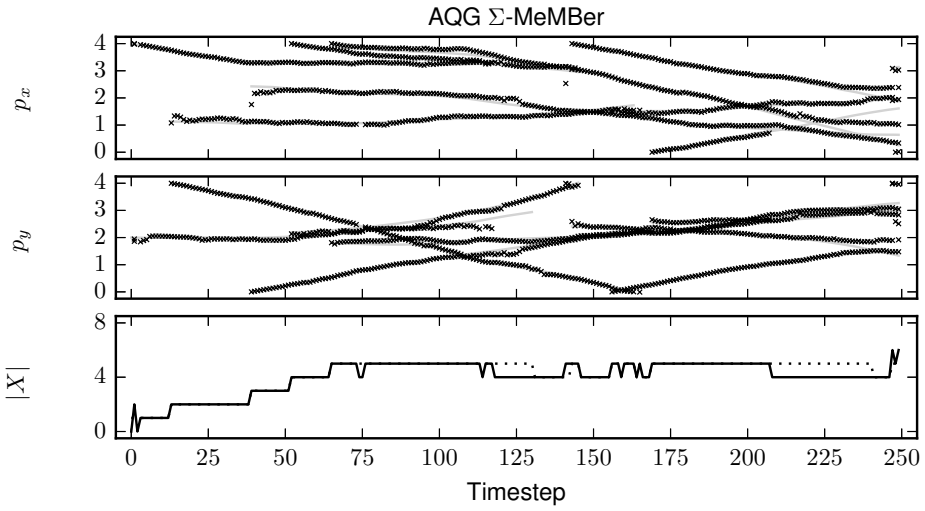


Figure B.6.: Exemplary tracking result for the Approximate Quasi Gaussian (AQG) $\Sigma$–MeMBer filter with $p_d = 1.0$. The dotted line is the ground truth and the solid line is the estimated track.

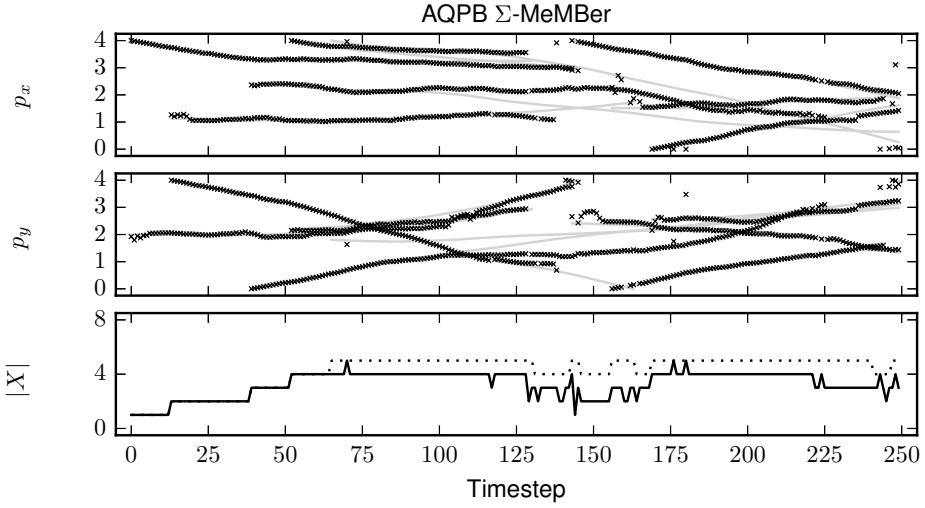


Figure B.7.: Exemplary tracking result for the IQPB CB-MeMBer filter with $p_d = 1.0$. The dotted line is the ground truth and the solid line is the estimated track.

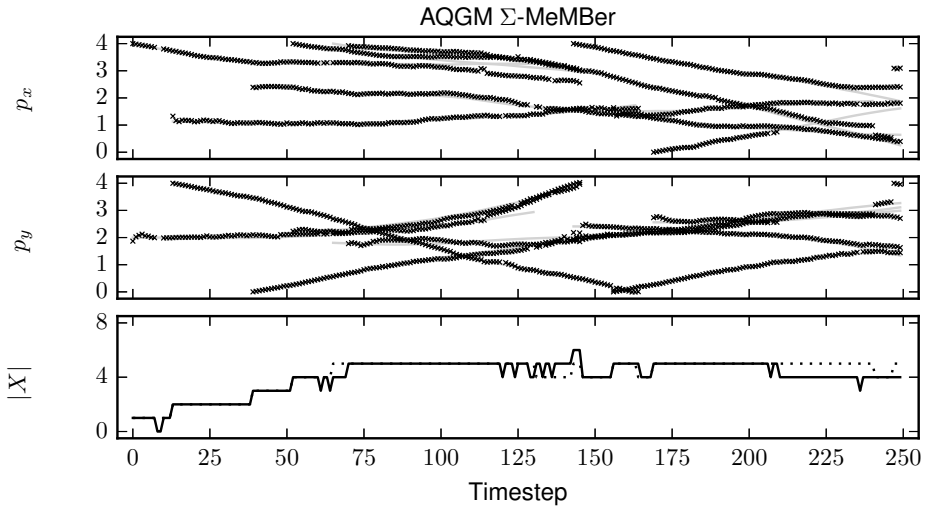


Figure B.8.: Exemplary tracking result for the Approximate Quasi Gaussian Mixture (AQGM) CB-MeMBer filter with $p_d = 1.0$. The dotted line is the ground truth and the solid line is the estimated track.

## B.2. Examples Track with Probability of Detection $p_d = 0.95$

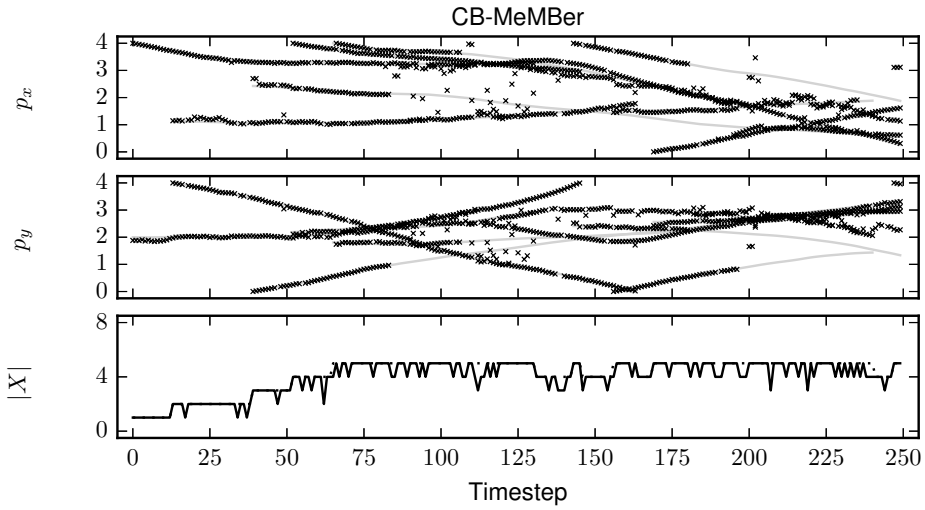

Figure B.9.: Exemplary tracking result for the CB-MeMBer filter with $p_d = 0.95$. The dotted line is the ground truth and the solid line is the estimated track.

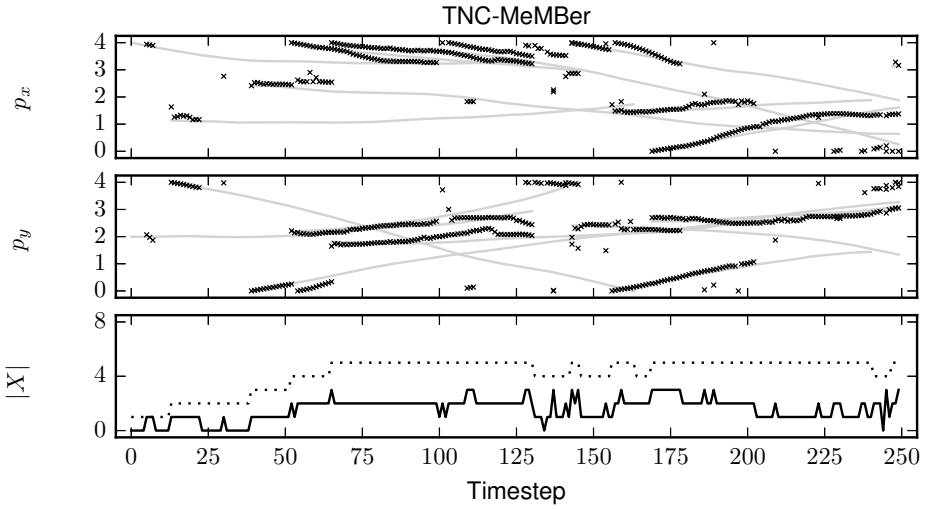


Figure B.10.: Exemplary tracking result for the TNC MeMBer filter with $p_d = 0.95$. The dotted line is the ground truth and the solid line is the estimated track.

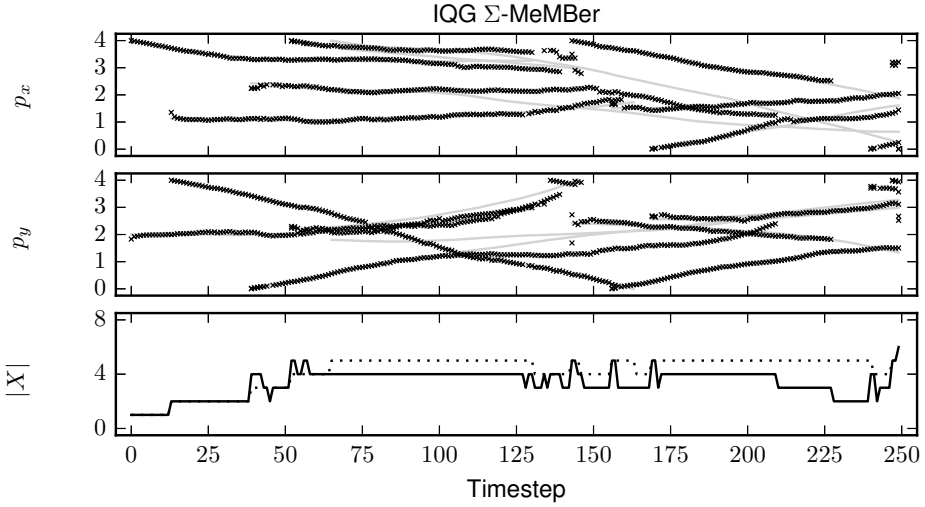


Figure B.11.: Exemplary tracking result for the IQG $\Sigma$–MeMBer filter with $p_d = 0.95$. The dotted line is the ground truth and the solid line is the estimated track.

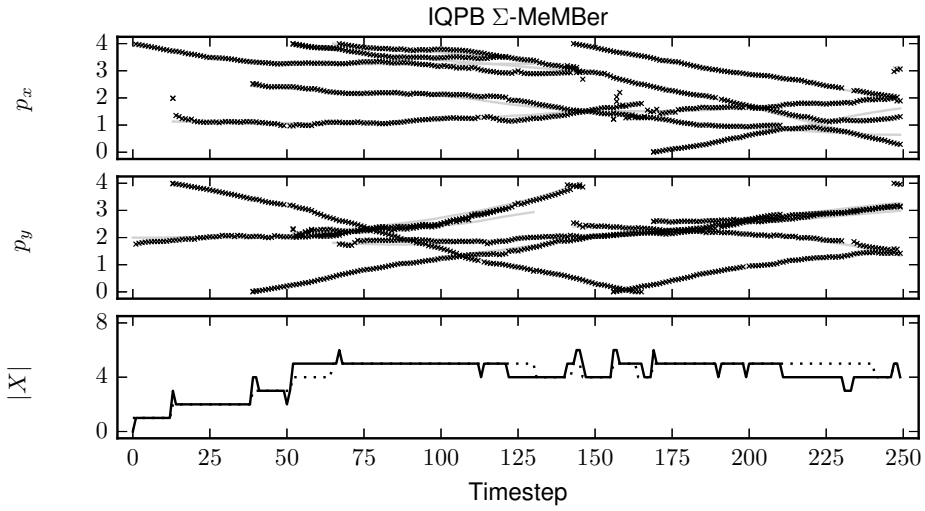


Figure B.12.: Exemplary tracking result for the IQPB CB-MeMBer filter with $p_d = 0.95$. The dotted line is the ground truth and the solid line is the estimated track.

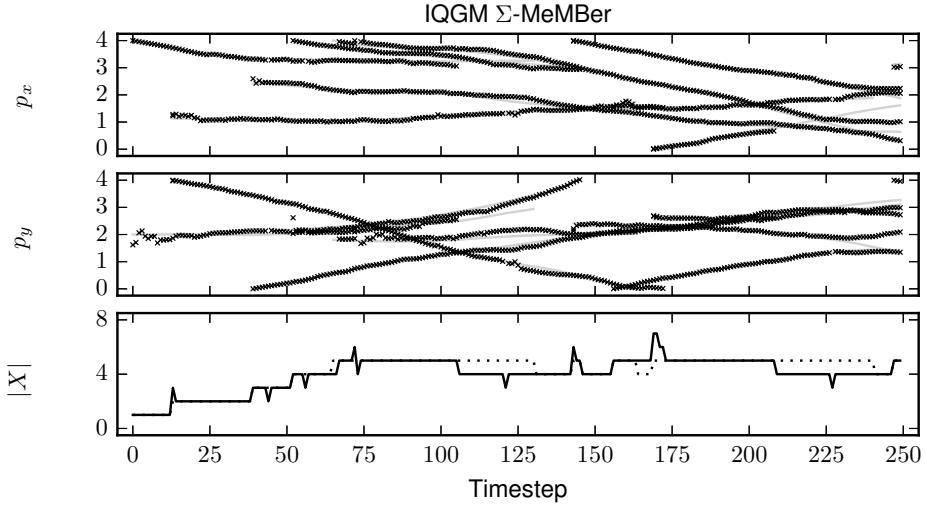


Figure B.13.: Exemplary tracking result for the IQGM CB-MeMBer filter with $p_d = 0.95$. The dotted line is the ground truth and the solid line is the estimated track.

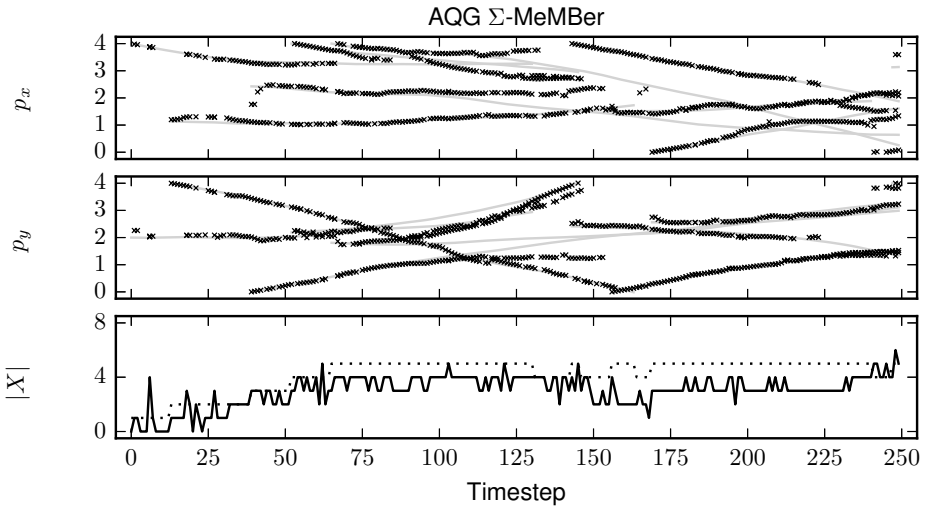


Figure B.14.: Exemplary tracking result for the AQG Σ–MeMBer filter with $p_d$ = 0.95. The dotted line is the ground truth and the solid line is the estimated track.

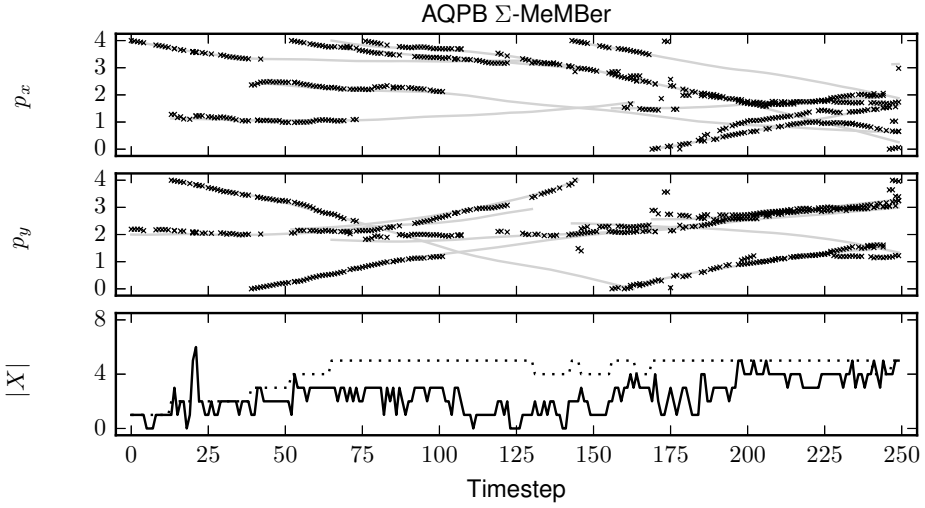


Figure B.15.: Exemplary tracking result for the IQPB CB-MeMBer filter with $p_d$ = 0.95. The dotted line is the ground truth and the solid line is the estimated track.

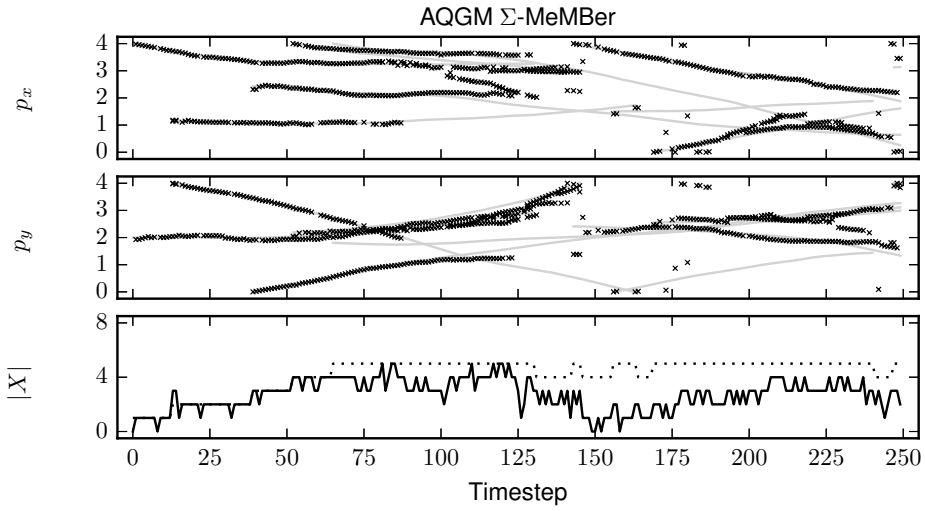


Figure B.16.: Exemplary tracking result for the AQGM CB-MeMBer filter with $p_d = 0.95$. The dotted line is the ground truth and the solid line is the estimated track.

## B.3. Examples Track with Probability of Detection $p_d = 0.9$

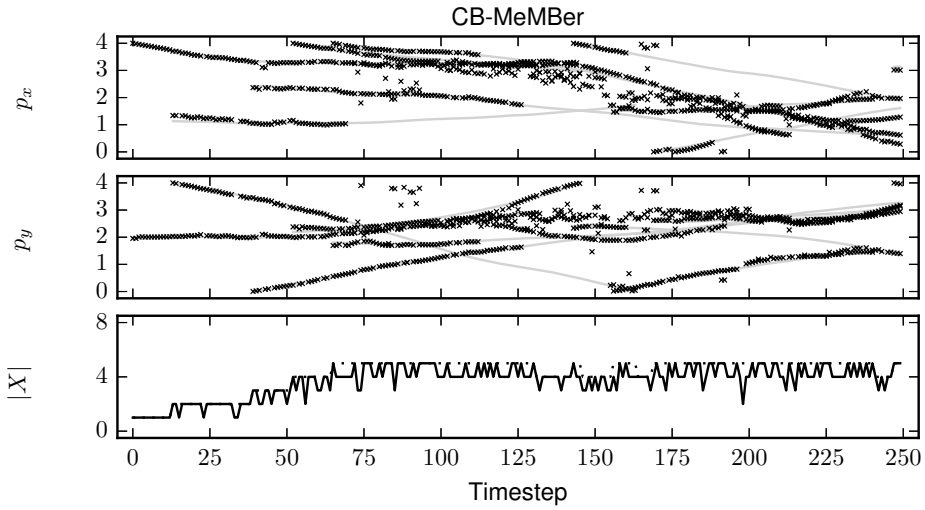

Figure B.17.: Exemplary tracking result for the CB-MeMBer filter with $p_d = 0.9$. The dotted line is the ground truth and the solid line is the estimated track.

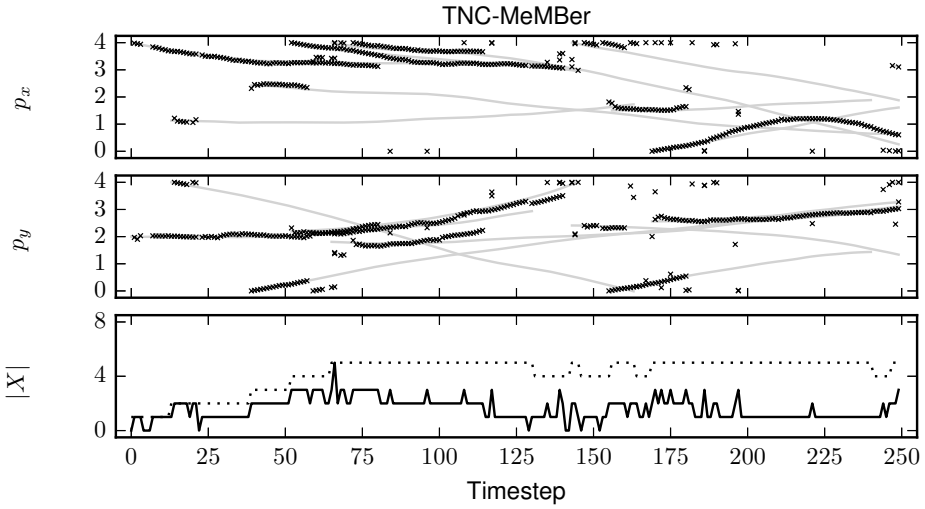


Figure B.18.: Exemplary tracking result for the TNC MeMBer filter with $p_d = 0.9$. The dotted line is the ground truth and the solid line is the estimated track.

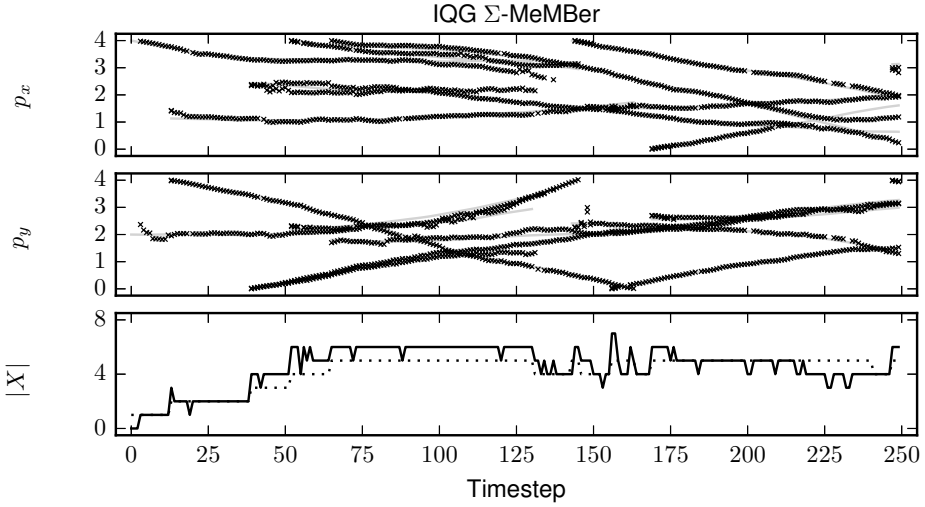


Figure B.19.: Exemplary tracking result for the IQG $\Sigma$–MeMBer filter with $p_d = 0.9$. The dotted line is the ground truth and the solid line is the estimated track.

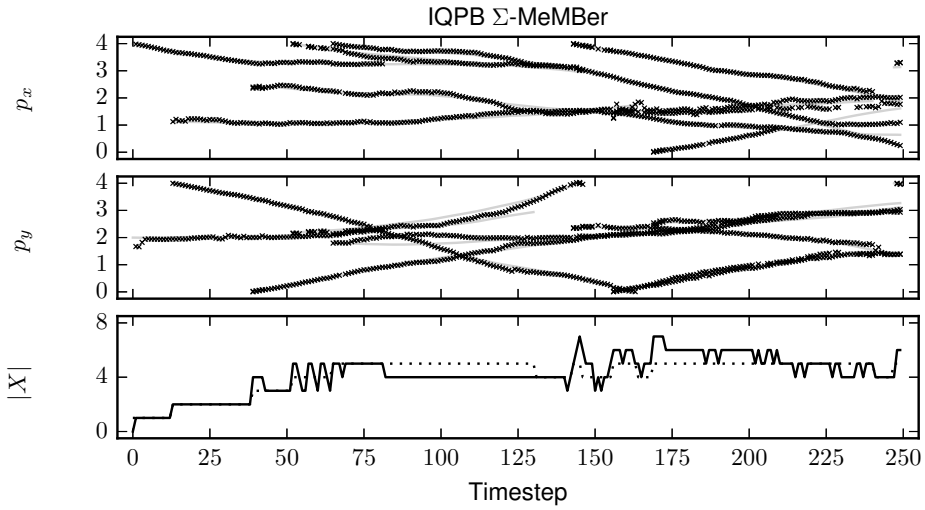


Figure B.20.: Exemplary tracking result for the IQPB CB-MeMBer filter with $p_d = 0.9$. The dotted line is the ground truth and the solid line is the estimated track.

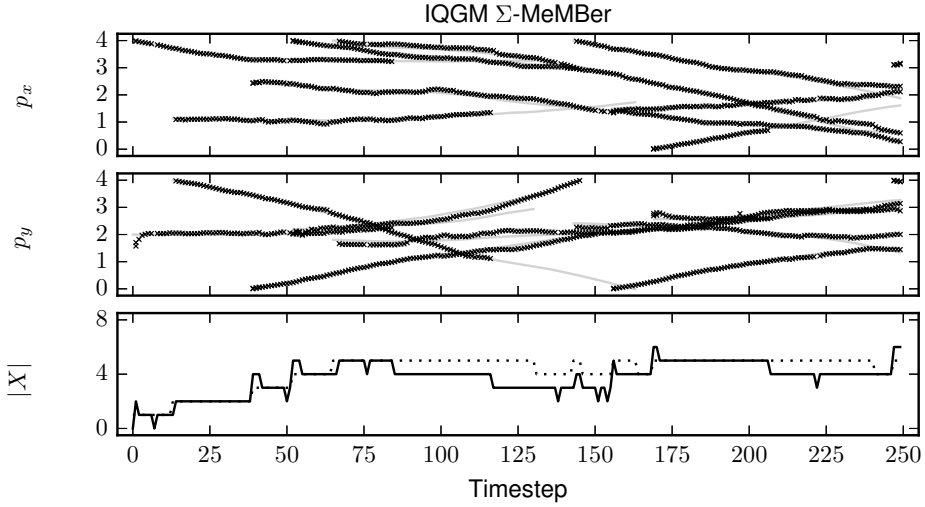


Figure B.21.: Exemplary tracking result for the IQGM CB-MeMBer filter with $p_d = 0.9$. The dotted line is the ground truth and the solid line is the estimated track.

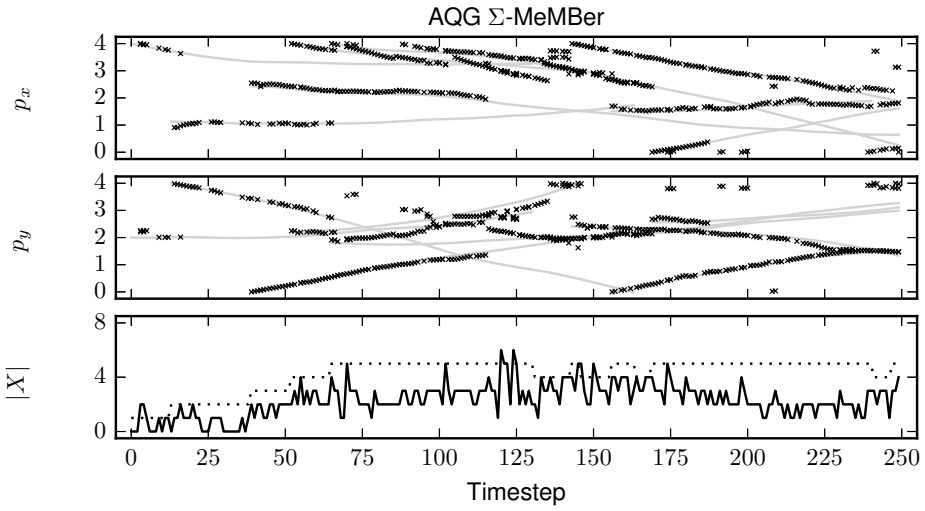


Figure B.22.: Exemplary tracking result for the AQG $\Sigma$–MeMBer filter with $p_d = 0.9$. The dotted line is the ground truth and the solid line is the estimated track.

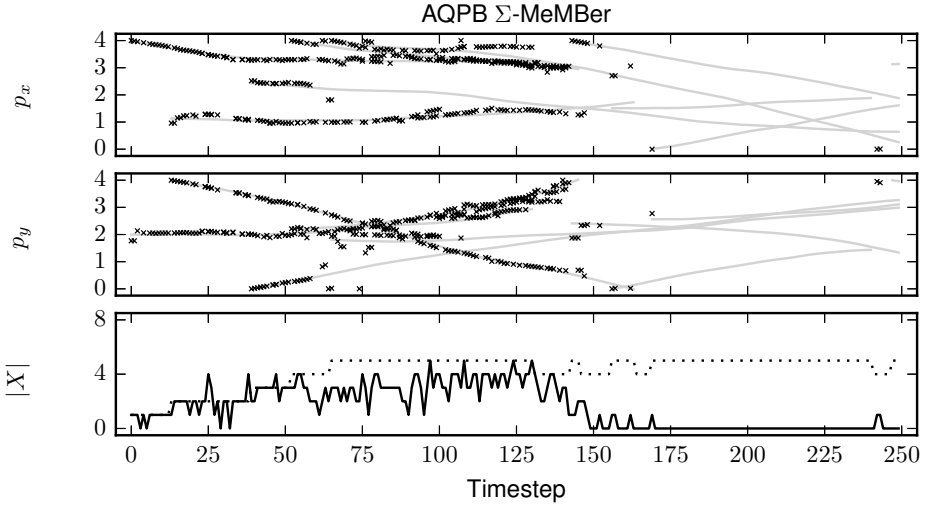


Figure B.23.: Exemplary tracking result for the IQPB CB-MeMBer filter with $p_d = 0.9$. The dotted line is the ground truth and the solid line is the estimated track.

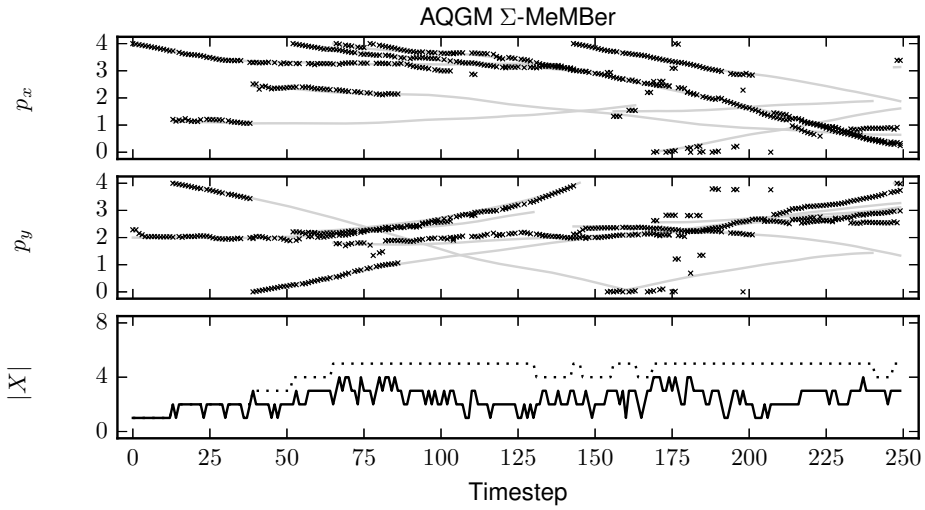


Figure B.24.: Exemplary tracking result for the AQGM CB-MeMBer filter with $p_d = 0.9$. The dotted line is the ground truth and the solid line is the estimated track.

## B.4. Examples Track with Probability of Detection $p_d = 0.8$

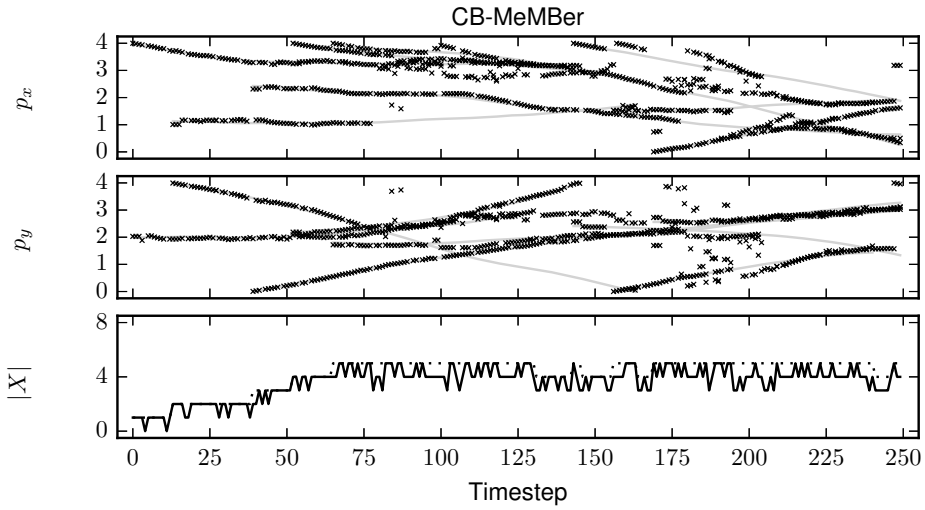

Figure B.25.: Exemplary tracking result for the CB-MeMBer filter with $p_d = 0.8$. The dotted line is the ground truth and the solid line is the estimated track.

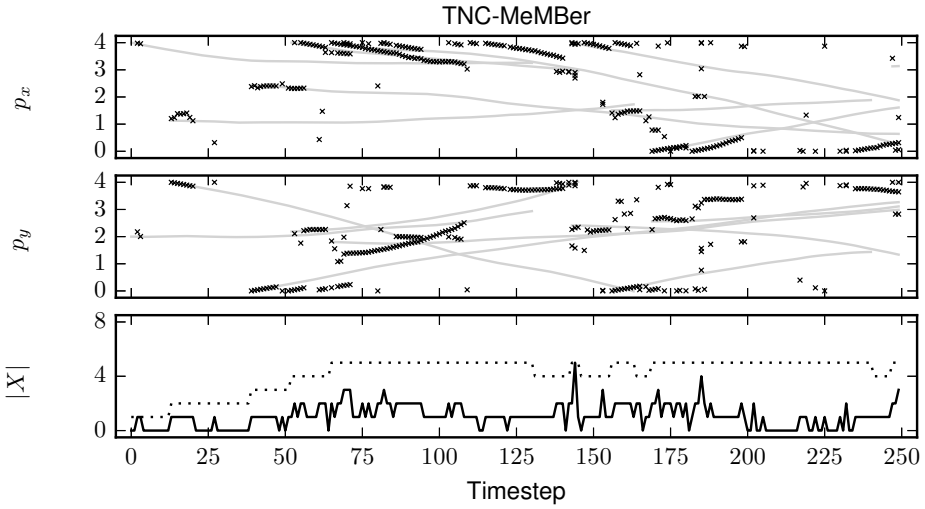


Figure B.26.: Exemplary tracking result for the TNC MeMBer filter with $p_d = 0.8$. The dotted line is the ground truth and the solid line is the estimated track.

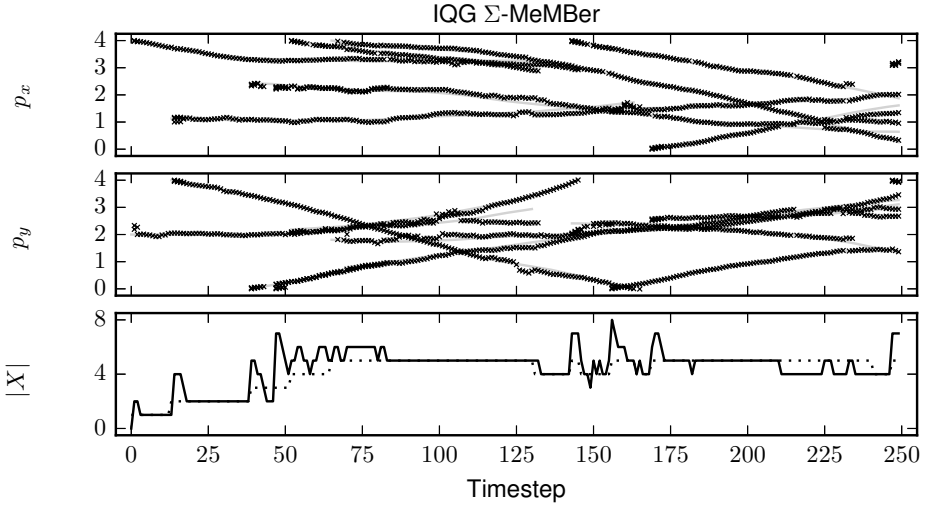


Figure B.27.: Exemplary tracking result for the IQG $\Sigma$–MeMBer filter with $p_d = 0.8$. The dotted line is the ground truth and the solid line is the estimated track.

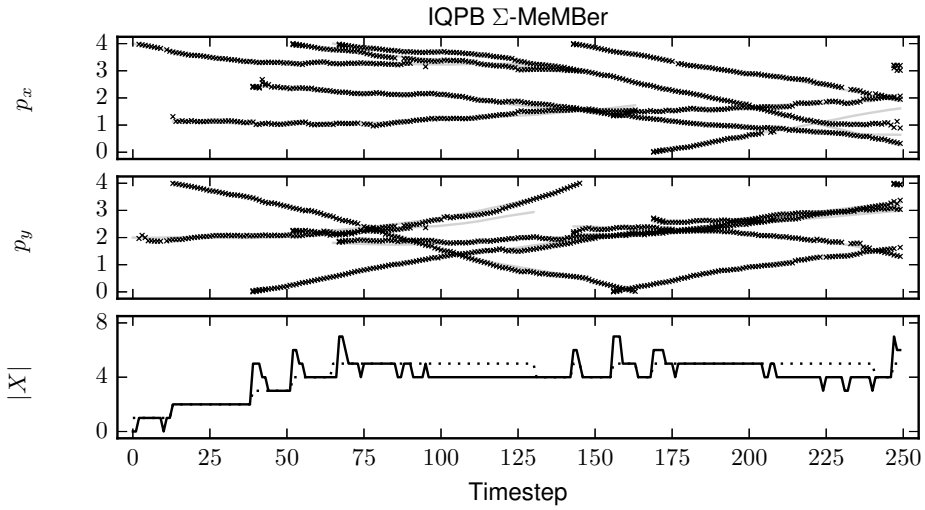


Figure B.28.: Exemplary tracking result for the IQPB CB-MeMBer filter with $p_d = 0.8$. The dotted line is the ground truth and the solid line is the estimated track.

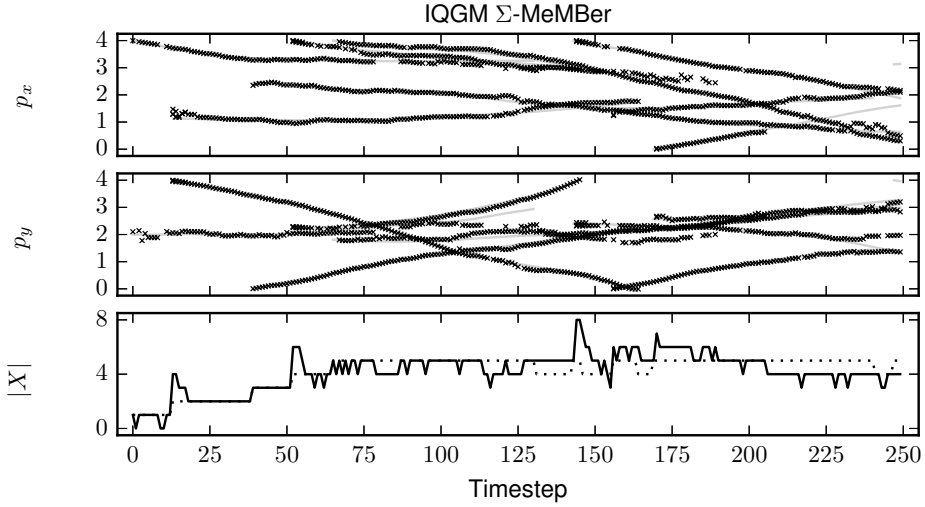


Figure B.29.: Exemplary tracking result for the IQGM CB-MeMBer filter with $p_d = 0.8$. The dotted line is the ground truth and the solid line is the estimated track.

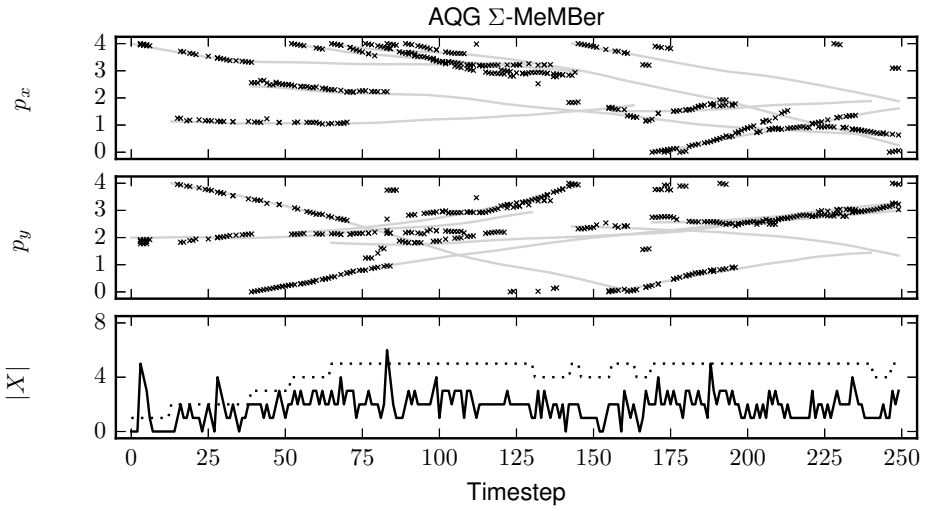


Figure B.30.: Exemplary tracking result for the AQG $\Sigma$–MeMBer filter with $p_d = 0.8$. The dotted line is the ground truth and the solid line is the estimated track.

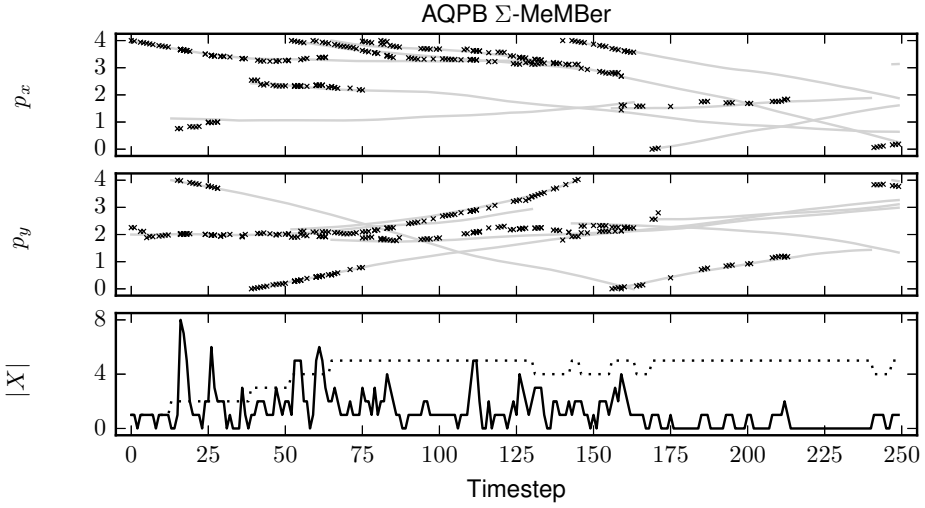


Figure B.31.: Exemplary tracking result for the IQPB CB-MeMBer filter with $p_d = 0.8$. The dotted line is the ground truth and the solid line is the estimated track.

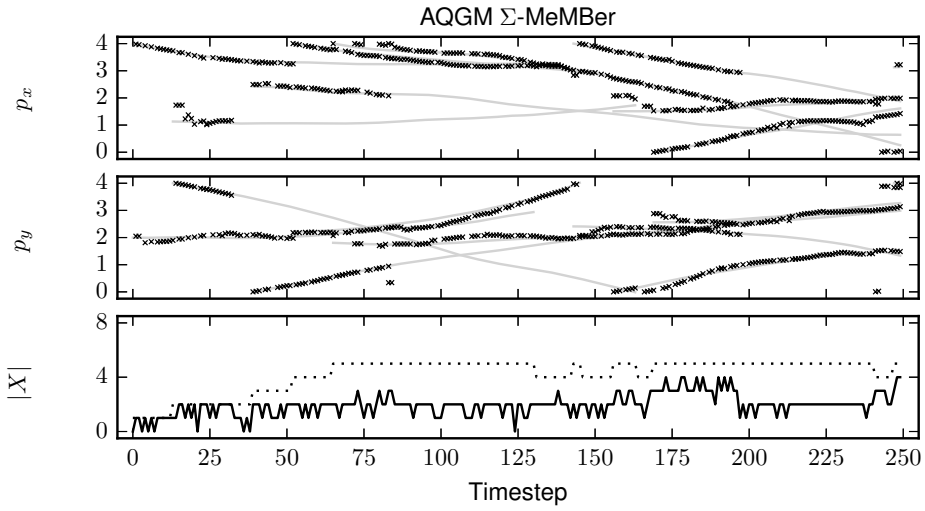


Figure B.32.: Exemplary tracking result for the AQGM CB-MeMBer filter with $p_d = 0.8$. The dotted line is the ground truth and the solid line is the estimated track.

## B.5. Examples Track with Probability of Detection $p_d = 0.7$

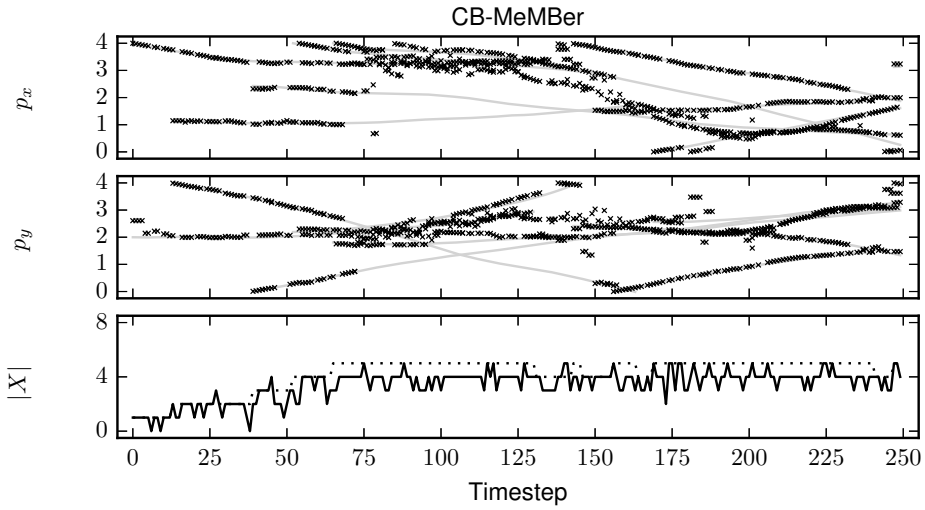

Figure B.33.: Exemplary tracking result for the CB-MeMBer filter with $p_d = 0.7$. The dotted line is the ground truth and the solid line is the estimated track.

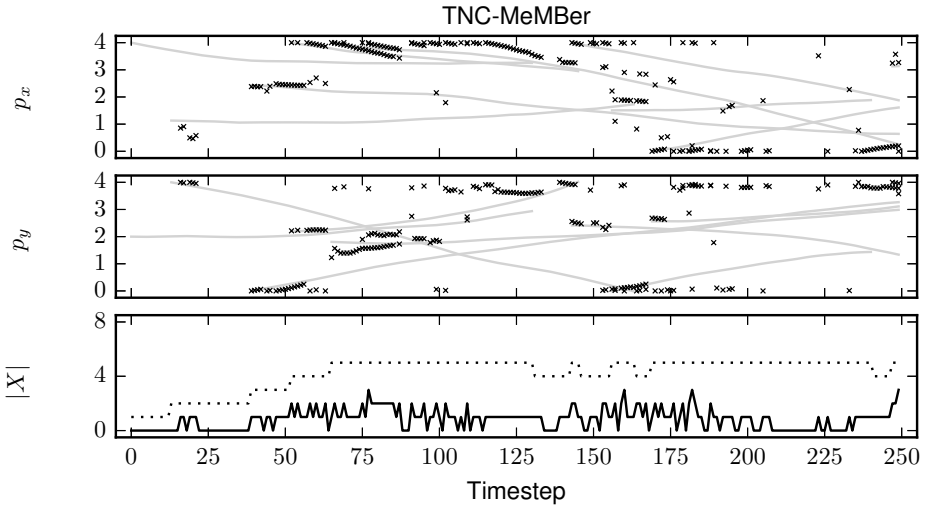


Figure B.34.: Exemplary tracking result for the TNC MeMBer filter with $p_d = 0.7$. The dotted line is the ground truth and the solid line is the estimated track.

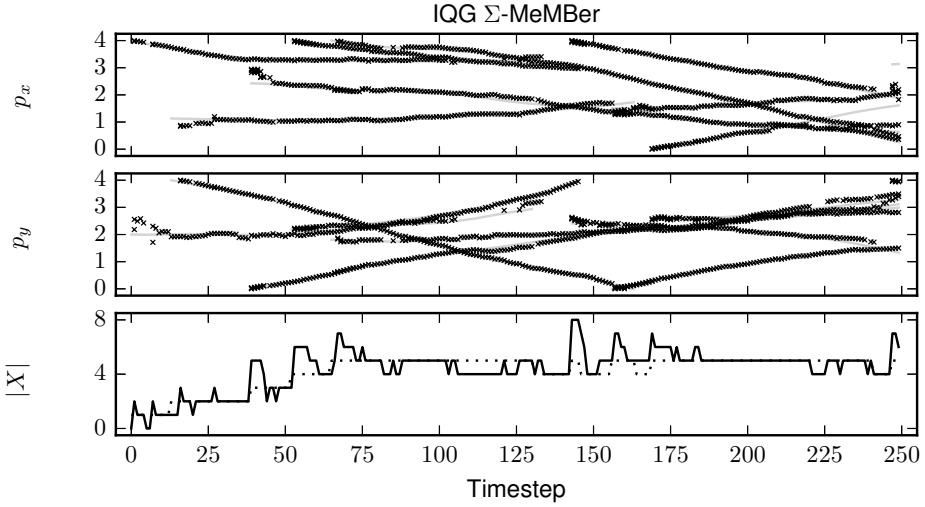


Figure B.35.: Exemplary tracking result for the IQG $\Sigma$–MeMBer filter with $p_d = 0.7$. The dotted line is the ground truth and the solid line is the estimated track.

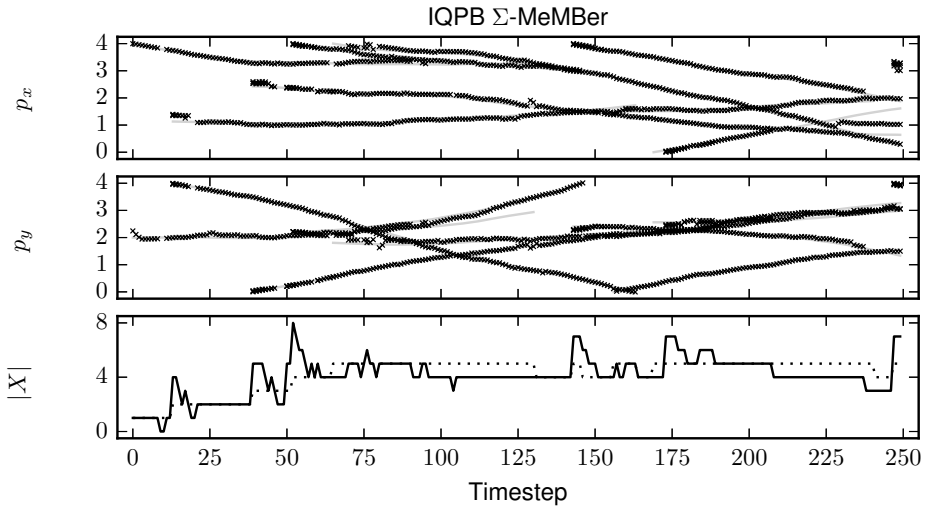


Figure B.36.: Exemplary tracking result for the IQPB CB-MeMBer filter with $p_d = 0.7$. The dotted line is the ground truth and the solid line is the estimated track.

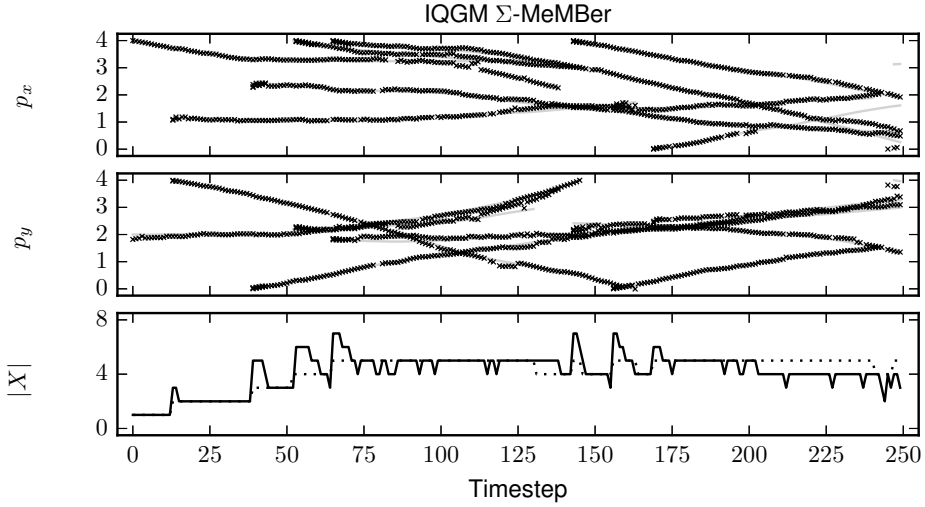


Figure B.37.: Exemplary tracking result for the IQGM CB-MeMBer filter with $p_d = 0.7$. The dotted line is the ground truth and the solid line is the estimated track.

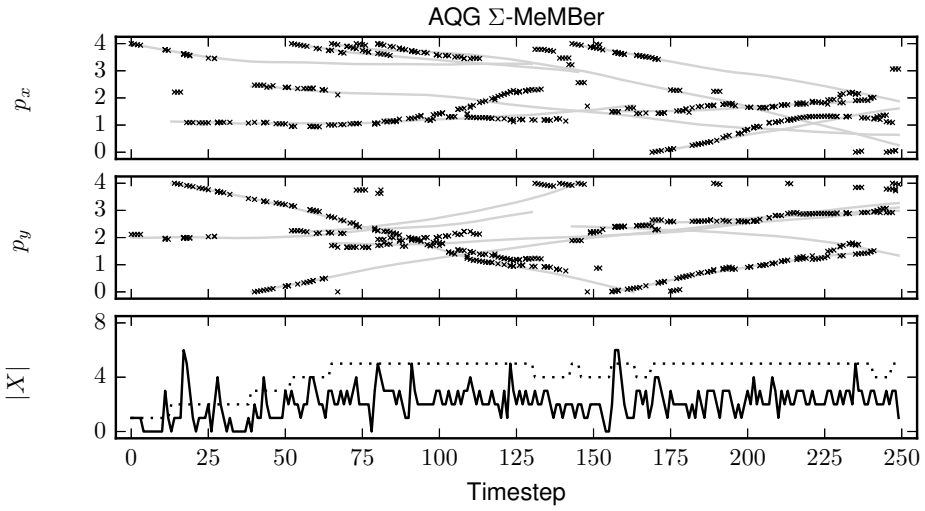


Figure B.38.: Exemplary tracking result for the AQG $\Sigma$–MeMBer filter with $p_d = 0.7$. The dotted line is the ground truth and the solid line is the estimated track.

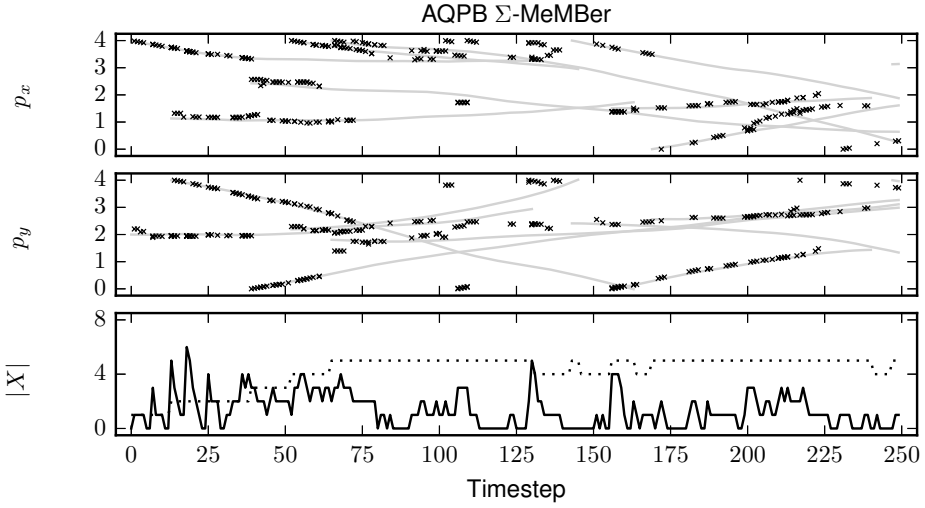


Figure B.39.: Exemplary tracking result for the IQPB CB-MeMBer filter with $p_d = 0.7$. The dotted line is the ground truth and the solid line is the estimated track.

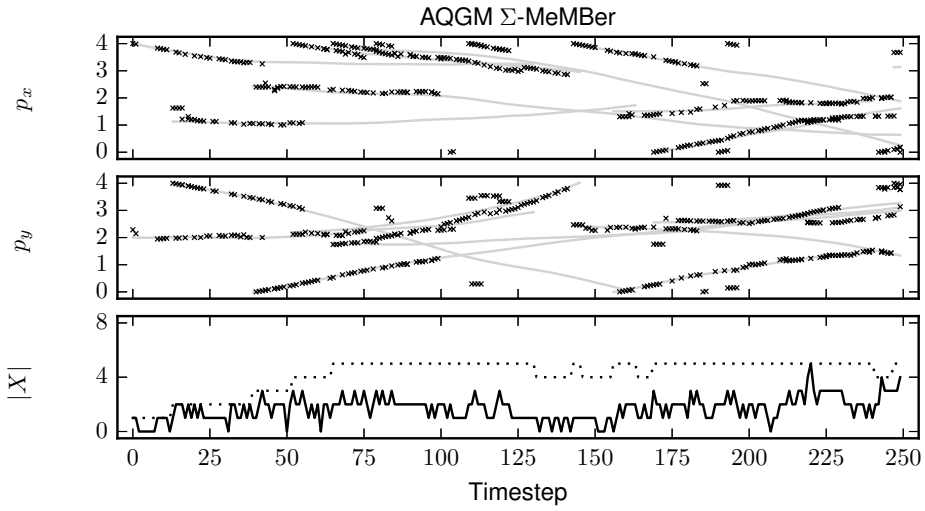


Figure B.40.: Exemplary tracking result for the AQGM CB-MeMBer filter with $p_d = 0.7$. The dotted line is the ground truth and the solid line is the estimated track.

## B.6. Examples Track with Probability of Detection $p_d = 0.6$

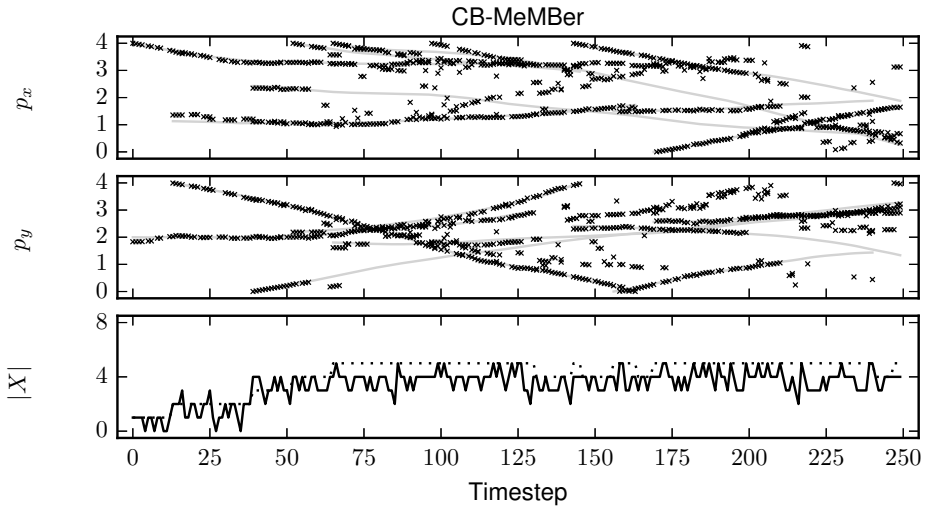

Figure B.41.: Exemplary tracking result for the CB-MeMBer filter with $p_d = 0.6$. The dotted line is the ground truth and the solid line is the estimated track.

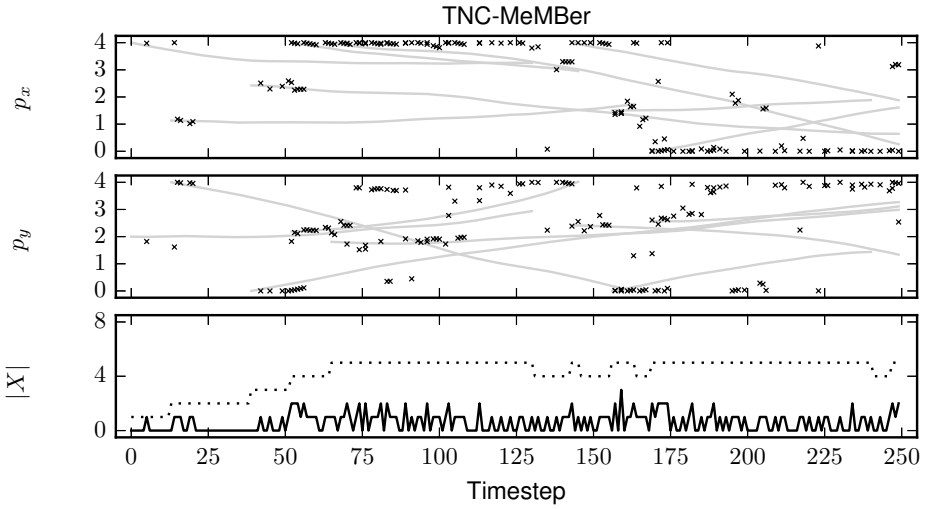


Figure B.42.: Exemplary tracking result for the TNC MeMBer filter with $p_d = 0.6$. The dotted line is the ground truth and the solid line is the estimated track.

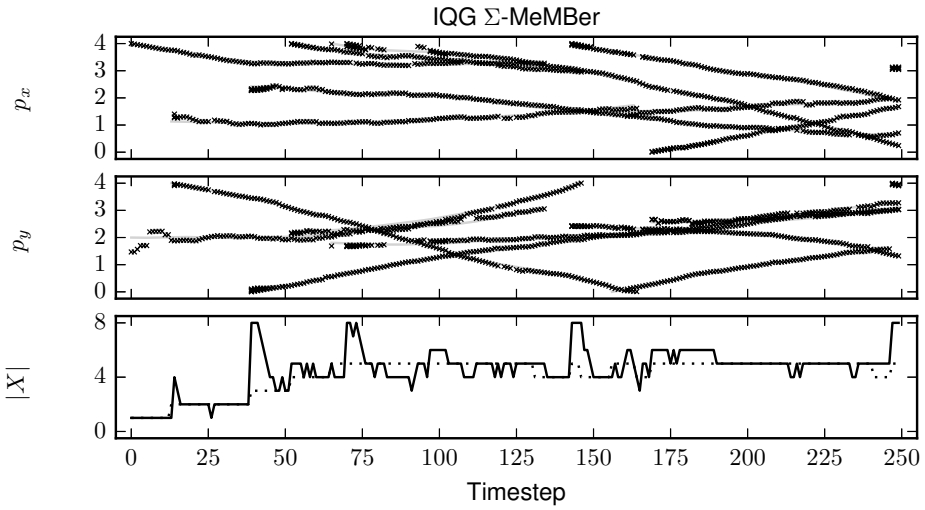


Figure B.43.: Exemplary tracking result for the IQG $\Sigma$–MeMBer filter with $p_d = 0.6$. The dotted line is the ground truth and the solid line is the estimated track.

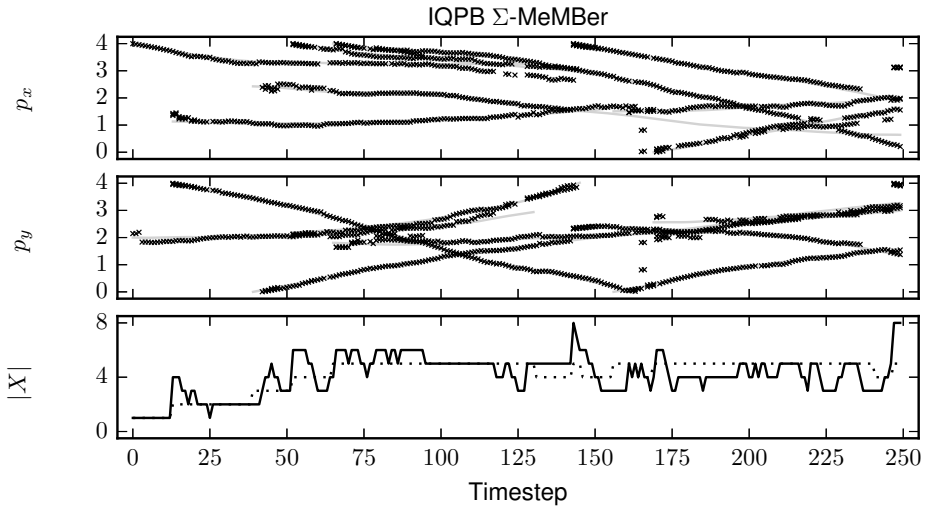


Figure B.44.: Exemplary tracking result for the IQPB CB-MeMBer filter with $p_d = 0.6$. The dotted line is the ground truth and the solid line is the estimated track.

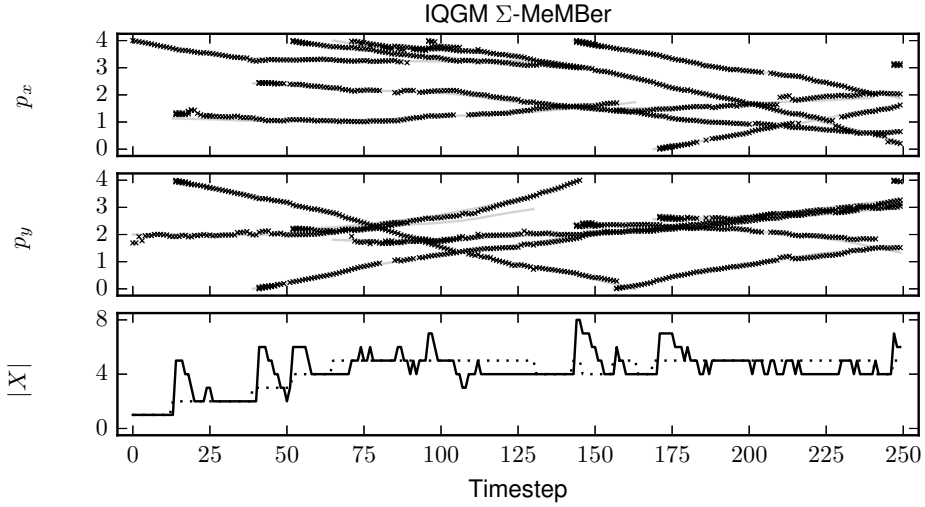


Figure B.45.: Exemplary tracking result for the IQGM CB-MeMBer filter with $p_d = 0.6$. The dotted line is the ground truth and the solid line is the estimated track.

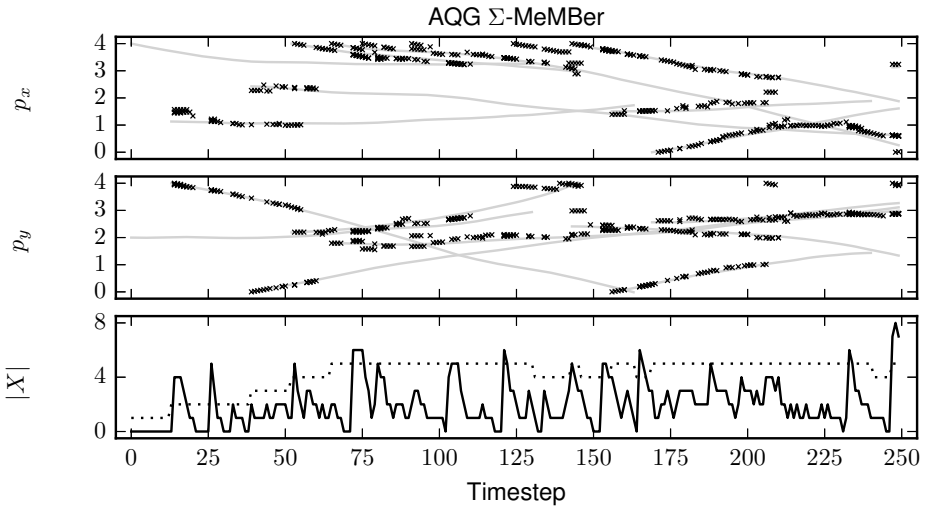


Figure B.46.: Exemplary tracking result for the AQG $\Sigma$–MeMBer filter with $p_d = 0.6$. The dotted line is the ground truth and the solid line is the estimated track.

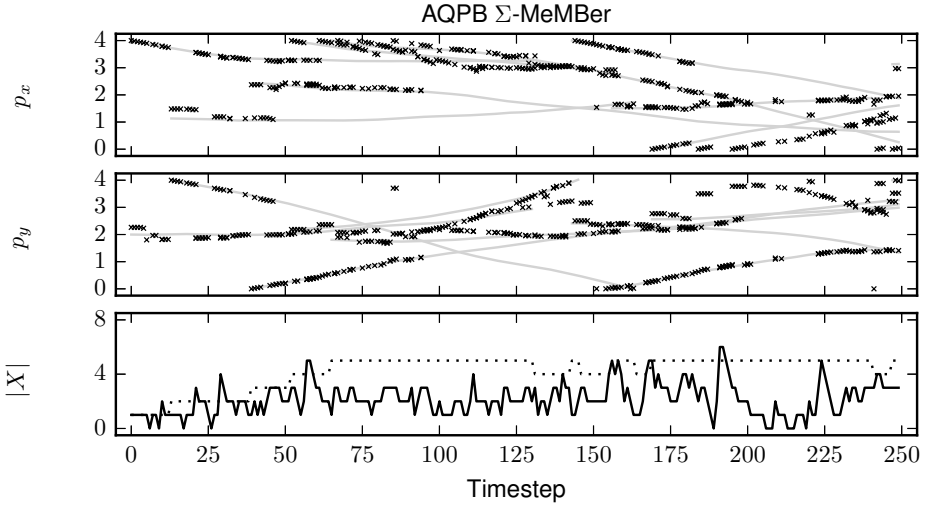


Figure B.47.: Exemplary tracking result for the IQPB CB-MeMBer filter with $p_d = 0.6$. The dotted line is the ground truth and the solid line is the estimated track.

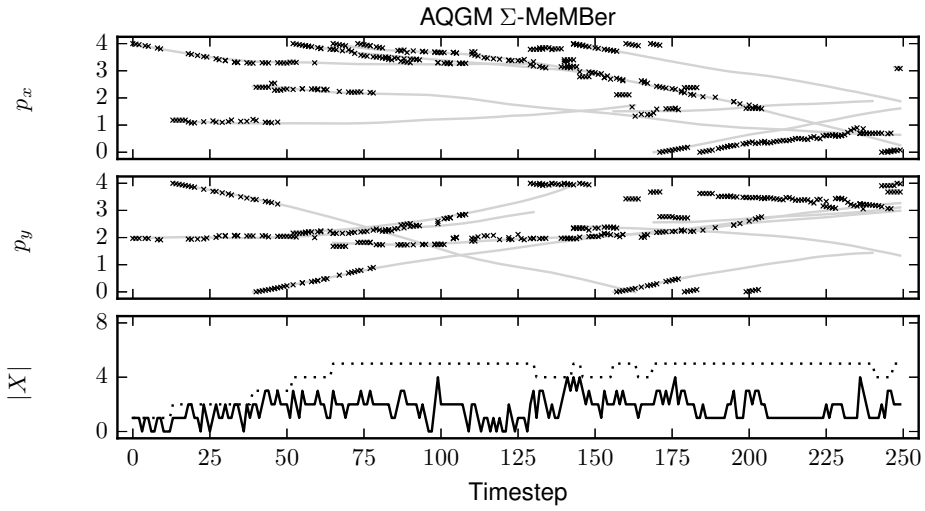


Figure B.48.: Exemplary tracking result for the AQGM CB-MeMBer filter with $p_d = 0.6$. The dotted line is the ground truth and the solid line is the estimated track.

## B.7. Examples Track with Probability of Detection $p_d = 0.5$

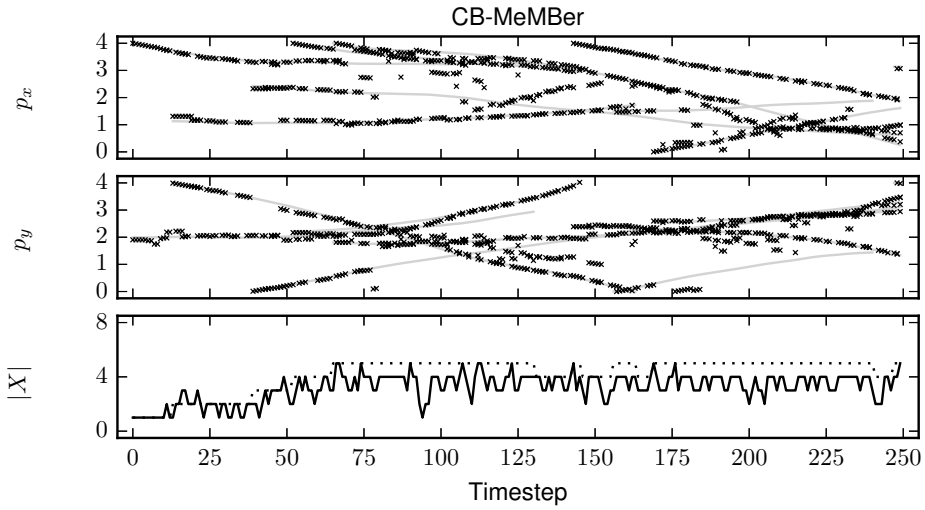

Figure B.49.: Exemplary tracking result for the CB-MeMBer filter with $p_d = 0.5$. The dotted line is the ground truth and the solid line is the estimated track.

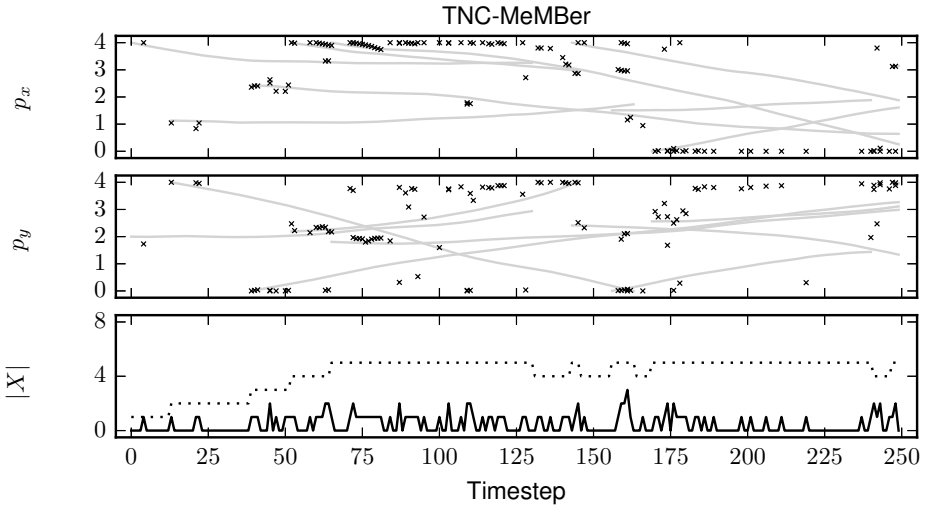


Figure B.50.: Exemplary tracking result for the TNC MeMBer filter with $p_d = 0.5$. The dotted line is the ground truth and the solid line is the estimated track.

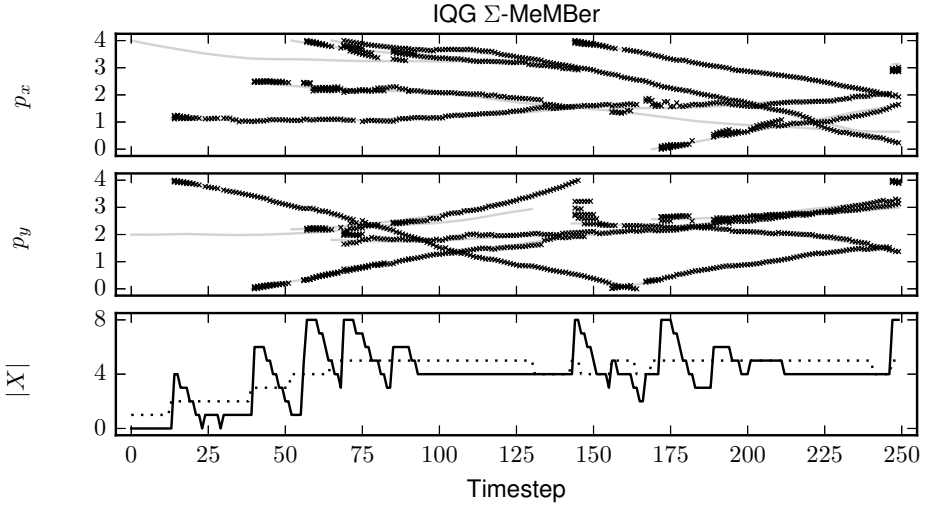


Figure B.51.: Exemplary tracking result for the IQG $\Sigma$–MeMBer filter with $p_d = 0.5$. The dotted line is the ground truth and the solid line is the estimated track.

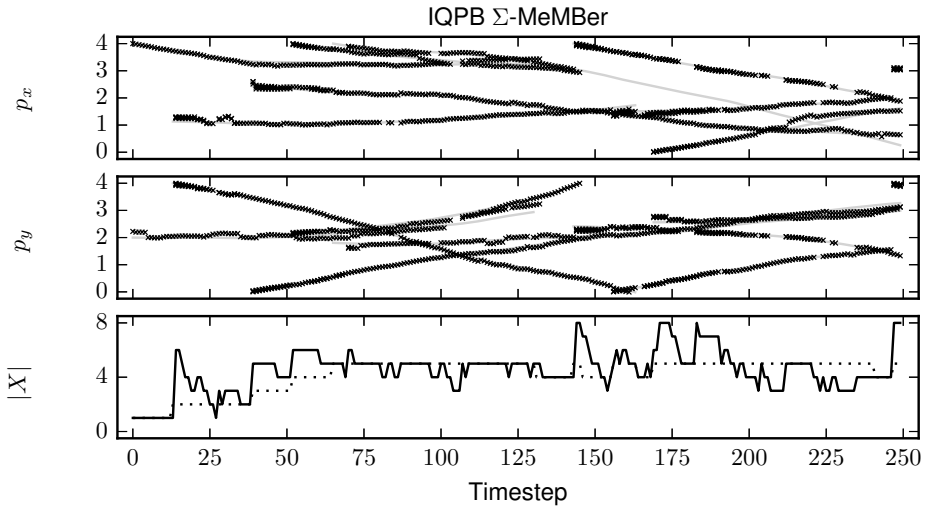


Figure B.52.: Exemplary tracking result for the IQPB CB-MeMBer filter with $p_d = 0.5$. The dotted line is the ground truth and the solid line is the estimated track.

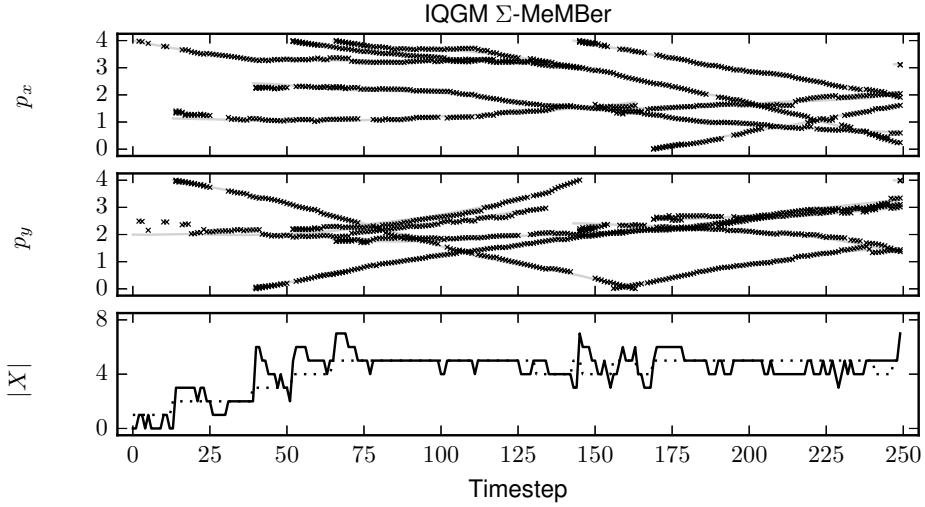


Figure B.53.: Exemplary tracking result for the IQGM CB-MeMBer filter with $p_d = 0.5$. The dotted line is the ground truth and the solid line is the estimated track.

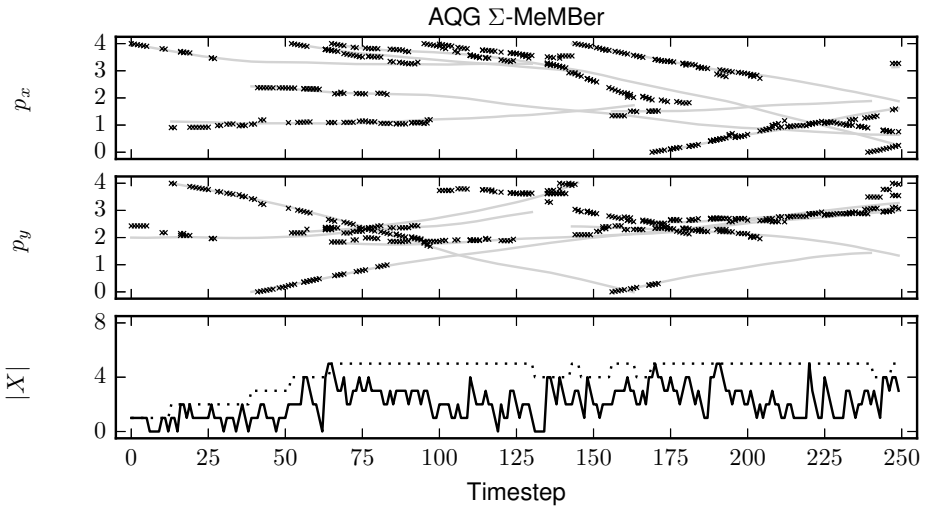


Figure B.54.: Exemplary tracking result for the AQG $\Sigma$–MeMBer filter with $p_d = 0.5$. The dotted line is the ground truth and the solid line is the estimated track.

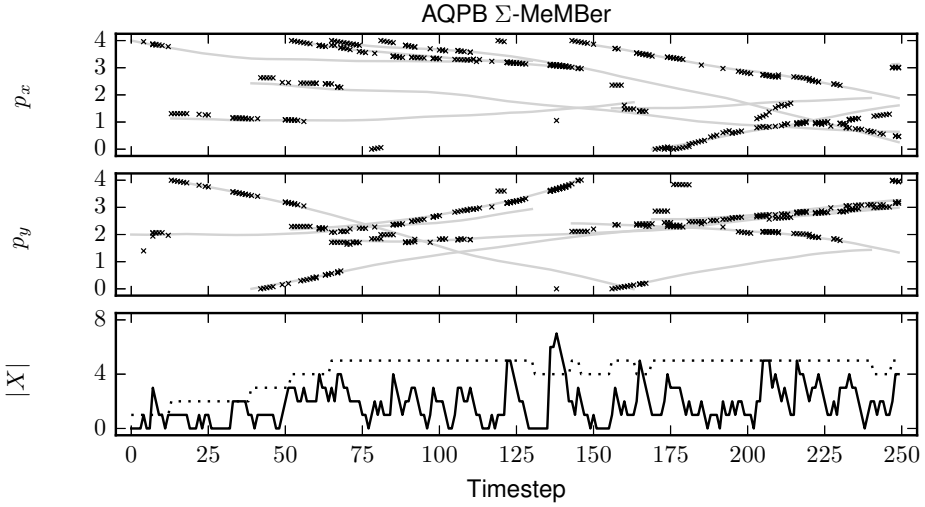


Figure B.55.: Exemplary tracking result for the IQPB CB-MeMBer filter with $p_d = 0.5$. The dotted line is the ground truth and the solid line is the estimated track.

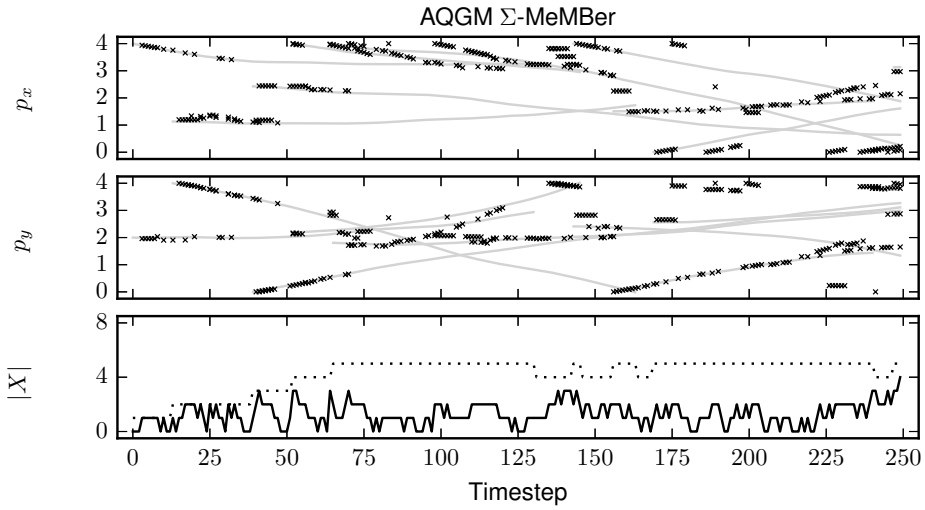


Figure B.56.: Exemplary tracking result for the AQGM CB-MeMBer filter with $p_d = 0.5$. The dotted line is the ground truth and the solid line is the estimated track.

# Publications

The following papers have been published by the author:


[Hau+10] D. Hauschildt, J. Kemper, N. Kirchhof, B. Juretko, and H. Linde. "Real-time scene simulator for Thermal Infrared Localization". In: *Proceedings of the 2010 Winter Simulation Conference (WSC)*. Dec. 2010, pp. 879–890. ISBN: 978-1-4244-9866-6. DOI: 10.1109/WSC.2010.5679101.

[Hau+12] D. Hauschildt, S. Kerner, S. Tasse, and O. Urbann. "Multi Body Kalman Filtering with Articulation Constraints for Humanoid Robot Pose and Motion Estimation". In: *RoboCup 2011: Robot Soccer World Cup XV*. Ed. by T. Röfer, N.M. Mayer, J. Savage, and U. Saranli. Vol. 7416. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, pp. 415–426. ISBN: 978-3-642-32059-0. DOI: 10.1007/978-3-642-32060-6_35.

[Hau11] D. Hauschildt. "Gaussian Mixture Implementation of the Cardinalized Probability Hypothesis Density Filter for Superpositional Sensors". In: *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. Guiamares, Sept. 2011, pp. 21–23. ISBN: 978-1-4577-1805-2. DOI: 10.1109/IPIN.2011.6071936.

[HK10] D. Hauschildt and N. Kirchhof. "Advances in thermal infrared localization: Challenges and solutions". In: *2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. Sept. 2010, pp. 1–8. ISBN: 978-1-4244-5862-2. DOI: 10.1109/IPIN.2010.5647415.

[HK11] D. Hauschildt and N. Kirchhof. "Improving indoor position estimation by combining active TDOA ultrasound and passive thermal infrared localization". In: *8th Workshop on Positioning, Navigation and Communication (WPNC)*. Apr. 2011, pp. 94–99. ISBN: 978-1-4577-0449-9. DOI: 10.1109/WPNC.2011.5961022.

[KH10] J. Kemper and D. Hauschildt. "Passive infrared localization with a Probability Hypothesis Density filter". In: *7th Workshop on Positioning Navigation and Communication (WPNC)*. Dresden, Mar. 2010, pp. 68–76. ISBN: 978-1-4244-7158-4. DOI: 10.1109/WPNC.2010.5653529.

[Pli+11] A. Plinge, D. Hauschildt, M.H. Hennecke, and G.A. Fink. "Multiple speaker tracking using a microphone array by combining auditory processing and a Gaussian mixture cardinalized probability hypothesis density filter". In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. May 2011, pp. 2476–2479. ISBN: 978-1-4577-0538-0. DOI: 10.1109/ICASSP.2011.5946986.

# Bibliography


[BF88] Yaakov. Bar-Shalom and Thomas E. Fortmann. *Tracking and data association / Yaakov Bar-Shalom, Thomas E. Fortmann*. Academic Press Boston, 1988. ISBN: 0120797607.

[Bla86] Samuel S. Blackman. *Multiple-target tracking with radar applications*. Artech House radar library. Norwood, Mass. Artech House, 1986. ISBN: 0-89006-179-3.

[DC05] R. Douc and O. Cappe. "Comparison of resampling schemes for particle filtering". In: *Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis (ISPA)*. Sept. 2005, pp. 64–69. DOI: 10.1109/ISPA.2005.195385.

[DV03] Daryl J. Daley and David Vere-Jones. *An introduction to the theory of point processes. vol. I. , Elementary theory and methods*. Probability and its applications. New York, Berlin, Paris: Springer, 2003. ISBN: 0-387-95541-0.

[Hau11] D. Hauschildt. "Gaussian Mixture Implementation of the Cardinalized Probability Hypothesis Density Filter for Superpositional Sensors". In: *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. Guiamares, Sept. 2011, pp. 21–23. ISBN: 978-1-4577-1805-2. DOI: 10.1109/IPIN.2011.6071936.

[Hon13] Yili Hong. "On comouting the distribution function for the Poisson binomial distribution". In: *Computational Statistics and Data Analysis* 59 (Mar. 2013), pp. 41–51. ISSN: 0167-9473. DOI: 10.1016/j.csda.2012.10.006..

[HVV11] Reza Hoseinnezhad, Ba-Ngu Vo, and Truong Nguyen Vu. "Visual Tracking of Multiple Targets by Multi-Bernoulli Filtering of Background Subtracted Image Data". English. In: *Advances in Swarm Intelligence*. Ed. by Ying Tan, Yuhui Shi, Yi Chai, and Guoyin Wang. Vol. 6729. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2011, pp. 509–518. ISBN: 978-3-642-21523-0. DOI: 10.1007/978-3-642-21524-7_63.

[KH10] J. Kemper and D. Hauschildt. "Passive infrared localization with a Probability Hypothesis Density filter". In: *7th Workshop on Positioning Navigation and Communication (WPNC)*. Dresden, Mar. 2010, pp. 68–76. ISBN: 978-1-4244-7158-4. DOI: 10.1109/WPNC.2010.5653529.

[Mac70] I. G. Macdonalds. *Symmetric Functions and Hall Polynomials*. Oxford Mathematical Monographs. Oxford Science Publications, 1970. ISBN: 0-19-853530-9.

[Mah03] Ronald Mahler. "Multitarget bayes filtering via first-order multitarget moments". In: *Ieee Transactions On Aerospace And Electronic Systems* 39.4 (Oct. 2003), pp. 1152–1178. ISSN: 0018-9251. DOI: 10.1109/TAES.2003.1261119.

[Mah04] Ronald Mahler. ""Statistics 101" for multisensor, multitarget data fusion". In: *Ieee Aerospace And Electronic Systems Magazine* 19.1 (Jan. 2004), pp. 53–64. ISSN: 0885-8985. DOI: 10.1109/MAES.2004.1263231.

[Mah07a] Ronald Mahler. "PHD filters of higher order in target number". In: *Ieee Transactions On Aerospace And Electronic Systems* 43.4 (2007), pp. 1523–1543. ISSN: 0018-9251. DOI: 10.1109/TAES.2007.4441756.

[Mah07b] Ronald Mahler. *Statistical Multisource–Multitarget Information Fusion*. Artech House, 2007. ISBN: 978-1-59693-092-6.

[Mah09] Ronald Mahler. "The multisensor PHD filter, I: General solution via multitarget calculus". In: ed. by Ivan Kadar. Vol. 7336. May 2009. DOI: 10.1117/12.818024.

[Mah13] Ronald Mahler. ""Statistics 102" for Multisource-Multitarget Detection and Tracking". In: *IEEE Journal of Selected Topics in Signal Processing* 7.3 (Mar. 2013), pp. 100–113. ISSN: 1932-4553. DOI: 10.1109/JSTSP.2013.2253084.

[Mah14] Ronald Mahler. *Advances in Statistical Multisource-Multitarget Information Fusion*. Artech House, 2014. ISBN: 978-1-60807-798-4.

[ME12] Ronald Mahler and Adel El-Fallah. "An approximate CPHD filter for superpositional sensors". In: *Proceedings of SPIE in Signal Processing, Sensor Fusion, and Target Recognition XXI*. Vol. 8392. May 2012. DOI: 10.1117/12.975965.

[NC13] S. Nannuru and M. Coates. "Multi-Bernoulli filter for superpositional sensors". In: *16th International Conference on Information Fusion (FUSION)*. Istanbul, July 2013, pp. 1632–1637. ISBN: 978-605-86311-1-3.

[NCM13] Santosh Nannuru, Mark Coates, and Ronald Mahler. "Computationally-tractable approximate PHD and CPHD filters for superpositional sensors". In: *IEEE Journal of Selected Topics in Signal Processing* 7.3 (May 2013), pp. 410–420. ISSN: 1932-4553. DOI: 10.1109/JSTSP.2013.2251605.

[RCV10] B. Ristic, D. Clark, and Ba-Ngu Vo. "Improved SMC implementation of the PHD filter". In: *13th Conference on Information Fusion (FUSION)*. Edinburgh, July 2010, pp. 1–8. ISBN: 978-0-9824438-1-1. DOI: 10.1109/ICIF.2010.5711922.

[Rei79] Donald B. Reid. "An Algorithm for Tracking Multiple Targets". In: *IEEE Transactions on Automatic Control* 24 (1979), pp. 843–854.

[Reu+14] Stephan Reuter, Ba-Ngu Vo, Ba-Tuong Vo, and Klaus Dietmayer. "The Labeled Multi-Bernoulli Filter". In: *IEEE Transactions on Signal Processing* (2014).

[Ris+13] B. Ristic, Ba-Tuong Vo, Ba-Ngu Vo, and A. Farina. "A Tutorial on Bernoulli Filters: Theory, Implementation and Applications". In: *IEEE Transactions on Signal Processing* 61.13 (2013), pp. 3406–3430. ISSN: 1053-587X. DOI: 10.1109/TSP.2013.2257765.

[RVC10] B. Ristic, Ba-Ngu Vo, and Daniel Clark. "Performance evaluation of multi-target tracking using the OSPA metric". In: *13th Conference on Information Fusion (FUSION)*. July 2010, pp. 1–7. ISBN: 978-0-9824438-1-1. DOI: 10.1109/ICIF.2010.5712055.

[TBF06] Sebastian Thrun, Wolfram Burghard, and Dieter Fox. *Probabilistic Robotics*. The MIT Press, 2006. ISBN: 978-0-2622-0162-9.

[Vo08] Ba Tuong Vo. "Random Finite Sets in Multi–Object Filtering". PhD thesis. School of Elektrical, Electronic and Computer Engineering at the University of Western Australia, Oct. 2008.

[VV13] Ba-Tuong Vo and Ba-Ngu Vo. "Labeled Random Finite Sets and Multi-Object Conjugate Priors". In: *IEEE Transactions on Signal Processing* 61.13 (July 2013), pp. 3460–3475. ISSN: 1053-587X. DOI: 10.1109/TSP.2013.2259822.

[VVC07] Ba-Tuong Vo, Ba-Ngu Vo, and Antonio Cantoni. "Analytic Implementations of the Cardinalized Probability Hypothesis Density Filter". In: *IEEE Transactions on Signal Processing* 55.7 (2007), pp. 3553–3567. ISSN: 1053587X. DOI: 10.1109/TSP.2007.894241.

[VVC09] Ba-Tuong Vo, Ba-Ngu Vo, and Antonio Cantoni. "The Cardinality Balanced Multi-Target Multi-Bernoulli Filter and Its Implementations". In: *IEEE Transactions on Signal Processing* 57.2 (2009), pp. 409–423. ISSN: 1053-587X. DOI: 10.1109/TSP.2008.2007924.

[Wan93] Y.H. Wang. "On the number of successes in independent trials". In: *Statistica Sinica* (1993), pp. 295–312.

[WSG07] Nick Whiteley, Sumeetpal Singh, and Simon Godsill. "Auxiliary Particle Implementation of the Probability Hypothesis Density Filter". In: *5th International Symposium on Image and Signal Processing and Analysis* 1.1 (Sept. 2007), pp. 510–515. DOI: 10.1109/ISPA.2007.4383746.