

THESIS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

Data association algorithms and metric design for trajectory estimation

ABU SAJANA RAHMATHULLAH



Department of Signals and Systems
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden
June 2016

Data association algorithms and metric design for trajectory estimation
ABU SAJANA RAHMATHULLAH
ISBN 978-91-7597-400-2

© ABU SAJANA RAHMATHULLAH, 2016.

Doktorsavhandlingar vid Chalmers tekniska högskola
Ny serie nr 4081
ISSN 0346-718X

Department of Signals and Systems
CHALMERS UNIVERSITY OF TECHNOLOGY
SE-412 96 Göteborg
Sweden
Telephone: +46 (0)31 – 772 1000

Typeset by the author using L^AT_EX.

Chalmers Reproservice
Göteborg, Sweden 2016

To everyone

Abstract

This thesis is concerned with trajectory estimation, which finds applications in various fields such as automotive safety and air traffic surveillance. More specifically, the thesis focuses on the data association part of the problem, for single and multiple targets, and on performance metrics.

Data association for single-trajectory estimation is typically performed using Gaussian mixture smoothing. To limit complexity, pruning or merging approximations are used. In this thesis, we propose systematic ways to perform a combination of merging and pruning for two smoothing strategies: forward-backward smoothing (FBS) and two-filter smoothing (TFS). We present novel solutions to the backward smoothing step of FBS and a likelihood approximation, called smoothed posterior pruning, for the backward filtering in TFS.

For data association in multi-trajectory estimation, we propose two iterative solutions based on expectation maximization (EM). The application of EM enables us to independently address the data association problems at different time instants, in each iteration. In the first solution, the best data association is estimated at each time instant using 2-D assignment, and given the best association, the states of the individual trajectories are immediately computed using Gaussian smoothing. In the second solution, we average the states of the individual trajectories over the data association distribution, which in turn is approximated using loopy belief propagation. Using simulations, we show that both solutions provide good trade-offs between accuracy and computation time compared to multiple hypothesis tracking.

For evaluating the performance of trajectory estimation, we propose two metrics that behave in an intuitive manner, capturing the relevant features in target tracking. First, the generalized optimal sub-pattern assignment metric computes the distance between finite sets of states, and addresses properties such as localization errors and missed and false targets, which are all relevant to target estimation. The second metric computes the distance between sets of trajectories and considers the temporal dimension of trajectories. We refine the concepts of track switches, which allow a trajec-

ABSTRACT

tory from one set to be paired with multiple trajectories in the other set across time, while penalizing it for these multiple assignments in an intuitive manner. We also present a lower bound for the metric that is remarkably accurate while being computable in polynomial time.

KEYWORDS: Trajectory estimation, data association, metrics, Gaussian mixtures, smoothing, expectation maximization

Acknowledgments

This thesis would not have been possible without the support from many. I would like to start by thanking Prof. Mats Viberg and Prof. Tomas McKelvey for giving me the opportunity to work with the Signal Processing group at Chalmers.

I would like to express my immense gratitude to my supervisor Lennart Svensson. You have been a great guide to me in several aspects, from producing this thesis to writing to skiing. I really appreciate your honest, constructive and elaborate feedbacks. I have learnt a lot from our technical discussions and from the way you can connect seemingly different problems. Thanks to your warm and friendly personality that made working with you such a pleasure.

Many thanks to my co-supervisor Daniel Svensson for his support over the years. I really appreciate that you have been there to help me with guidance and feedbacks. I have also enjoyed all the discussions I have had with you.

Thanks also to my co-supervisor Ángel F. García-Fernández for the exhilarating discussions, prompt feedbacks and for being a great host during my two months stay in Australia.

I want to thank my collaborator/Masters thesis student Raghavendra Selvan for all the valuable discussions we have had. Thanks for helping me with the papers for this thesis despite short notice.

I would also like to thank Prof. Ba-Ngu Vo for allowing me to visit his group at Curtin University. My sincere thanks to Karl, Samuel and Erik for their valuable time in proofreading this thesis. Thanks to all my current and former colleagues at Chalmers—Maryam, Lars, Ayca, Anders, Malin, Astrid, Naga, Keerthi, Sathya, Srikar, Rahul—who have made the work place such a relaxed environment. I have enjoyed my five years at Chalmers that would not have been possible without you all.

Thanks to all my friends who add a lot of fun to my life outside Chalmers. Thanks go to Ninva, Dipti, Gokul, Sanjna, Sandra, Deepthi, Annelie, Raneesh, Jagadeep, Aravindh and Kavitha.

I cannot thank enough my family—Zerina, Rahmathullah, Ziana, Asan

ACKNOWLEDGMENTS

and Hussain—who have always been an immense support for me. Thanks for the unconditional love you have for me. Last but not the least, thanks to Rickard for his love and support. Thanks for being there during the ups and downs of my life.

Publications

This thesis is based on the following publications:

Paper I

A.S. Rahmathullah, L. Svensson and D. Svensson, “Merging-based forward-backward smoothing on Gaussian mixtures”, In *Proc. 17th International Conference on Information Fusion (FUSION)*, Salamanca, Spain, July 2014.

Paper II

A.S. Rahmathullah, L. Svensson and D. Svensson, “Two-filter Gaussian mixture smoothing with posterior pruning”, In *Proc. 17th International Conference on Information Fusion (FUSION)*, Salamanca, Spain, July 2014.

Paper III

A.S. Rahmathullah, R. Selvan and L. Svensson, “A batch algorithm for estimating trajectories of point targets using expectation maximization”, Accepted for publication in *IEEE Transactions on Signal Processing*, April 2016.

Paper IV

R. Selvan, A.S. Rahmathullah and L. Svensson, “Expectation maximization with hidden state variables for multi-target tracking”, To be submitted to *IEEE Transactions on Signal Processing*.

Paper V

A.S. Rahmathullah, Á.F. García-Fernández and L. Svensson, “Generalized optimal sub-pattern assignment metric”, Submitted to *IEEE Transactions on Signal Processing*, Jan 2016.

Paper VI

A.S. Rahmathullah, Á.F. García-Fernández and L. Svensson, “A metric on the space of finite sets of trajectories for evaluation of multi-target tracking algorithms”, Submitted to *IEEE Transactions on Signal Processing*, May 2016.

Contents

Abstract	iii
Acknowledgments	v
Publications	vii
Contents	ix
I Introductory chapters	xv
1 Introduction	1
2 Models, objectives and conceptual solution	5
2.1 State space representation	5
2.2 Problem statement and conceptual solution	6
2.2.1 Prediction and filtering	6
2.2.2 Smoothing	7
2.3 Models	8
3 Single trajectory estimation	11
3.1 Linear and Gaussian filtering and smoothing	11
3.1.1 Kalman filtering	11
3.1.2 Smoothing	12
3.2 Non-linear models	13
3.2.1 Gaussian filters	14
3.2.2 Particle filters	15
3.3 Gaussian mixture filtering and smoothing	16
3.3.1 Optimal solution	17
3.4 Gaussian mixture reduction	19
3.4.1 Pruning	19
3.4.2 Merging	19
3.4.3 Choice of GMR	20

3.4.4	GMR for FBS and TFS	20
4	Multi-trajectory estimation	23
4.1	Data association	23
4.2	Tracking algorithms	24
4.2.1	Global nearest neighbour	25
4.2.2	Multiple hypothesis tracking	25
4.2.3	Joint probabilistic data association	26
4.2.4	Probabilistic multiple hypothesis tracking	26
4.3	EM for data association	26
5	Metrics	29
5.1	Need for a metric	29
5.2	Metric properties	30
5.3	Common metrics	31
5.3.1	On vector spaces	31
5.3.2	On the space of finite sets of vectors	32
5.3.3	On the space of finite sets of trajectories	33
6	Contributions and future work	37
6.1	Contributions	37
6.2	Future work	40
	References	43
II	Appended papers	53
Paper I	Merging-based forward-backward smoothing on Gaussian mixtures	57
1	Introduction	57
2	Problem formulation and idea	59
2.1	Idea	60
3	Background	61
3.1	Forward-Backward Smoothing	62
3.2	Optimal Gaussian mixture FBS	62
3.3	Forward filter based on pruning and merging approximations	63
4	Backward smoothing through merged nodes	65
4.1	Notation	67
4.2	Smoothing on merged nodes	69
5	Algorithm description	70

CONTENTS

6	Implementation and simulation Results	73
6.1	Simulation scenario	73
6.2	Implementation details	73
6.3	Results	74
7	Conclusion	75
A	Weight calculation	76
	References	77
Paper II Two-filter Gaussian mixture smoothing with posterior pruning		81
1	Introduction	81
2	Problem statement and idea	83
2.1	Idea	83
3	Background	84
3.1	Two-filter smoothing	84
4	Intragroup approximations of the backward likelihood	86
4.1	Structure of the backward likelihood	86
4.2	Normalizability of the BL and intragroup approximations	87
5	Smoothed posterior-based pruning	90
5.1	Posterior-based pruning	90
6	Algorithm	94
7	Implementation and simulation results	94
7.1	Simulation scenario	94
7.2	Implementation details	94
7.3	Results	97
8	Conclusion	99
	References	99
Paper III A batch algorithm for estimating trajectories of point targets using expectation maximization		103
1	Introduction	103
2	Problem formulation and background	106
2.1	Nomenclature	106
2.2	Problem formulation	107
2.3	Background	108
3	EM for target tracking	110
3.1	Expectation maximization for MAP estimation	110
3.2	Derivation of the EM algorithm for tracking multiple point targets	112
4	Marginal data association probabilities calculation using Loopy belief propagation	115

CONTENTS

4.1	Loopy belief propagation algorithm	116
5	Algorithm	118
6	Comparison with PMHT and JPDA	119
6.1	Comparison with PMHT	120
6.2	Comparison with JPDA	122
7	Simulations and results	125
7.1	Simulation scenarios	125
7.2	Handling sensitivity to initialization	127
7.3	Results	130
8	Conclusions and future work	132
A	Derivation of the EM algorithm for the Gaussian case	133
B	Smoothing with the RTS smoother	134
	References	134

Paper IV Expectation maximization with hidden state variables for multi-target tracking 141

1	Introduction	141
2	Problem formulation and background	144
2.1	Problem formulation	144
2.2	Background	145
3	EM for multi-target tracking	148
3.1	Introduction to expectation maximization	148
3.2	Derivation for data association estimation using EM .	149
3.3	Discussion	153
4	Algorithm	154
4.1	Description	154
4.2	Initialization	154
5	Track hypothesis switching	156
5.1	Comparing track hypotheses	158
5.2	Detecting targets in close proximity	159
6	Simulation and results	160
6.1	Simulations	160
6.2	Results	161
7	Conclusion and future work	168
A	Derivation of the M-step of EM algorithm	169
B	Smoothing with the RTS smoother	170
C	Modified auction algorithm	171
	References	171

Paper V Generalized optimal sub–pattern assignment metric 179

1	Introduction	179
2	Background	181
	2.1 Metric	182
	2.2 OSPA metric	182
	2.3 Drawback with the normalization in OSPA	184
3	Generalized OSPA metric	184
	3.1 Discussion	185
4	Choice of α in MTT	186
	4.1 Motivating the choice of $\alpha = 2$	186
	4.2 Reformulating GOSPA for $\alpha = 2$	188
5	Performance evaluation of MTT algorithms	189
6	Illustrations	192
7	Conclusions	194
A	Proof of the triangle inequality of GOSPA	195
B	Proof of the triangle inequality of OSPA	198
C	Proof of the average GOSPA metric	199
	References	201

Paper VI A metric on the space of finite sets of trajectories for evaluation of multi–target tracking algorithms 207

1	Introduction	207
2	Problem formulation and background	210
	2.1 Fundamental properties	210
	2.2 Challenges	211
	2.3 Bento’s natural and computable metrics	212
3	Assignment metric	214
	3.1 Overview	215
	3.2 Multi-dimensional assignment metric	215
	3.3 Interpretation	217
	3.4 Computation	218
4	LP relaxation metric	219
	4.1 Binary linear programming formulation	219
	4.2 LP relaxation metric	220
	4.3 Implementation of the LP relaxation metric using ADMM	221
5	Extension to random sets of trajectories	222
6	Simulation results	223
	6.1 Comparison between ADMM and ‘linprog’ implementations	223
	6.2 Example in MTT using sets of trajectories	225
7	Conclusion	228

CONTENTS

A	Proof for Proposition VI.9	229
A.1	Proof for computability using LP	229
A.2	Proof for metric properties	229
B	Formulation of the LP metric in consensus form	235
	References	237

Part I

Introductory chapters

Chapter 1

Introduction

In many applications, the objective is to systematically and sequentially estimate quantities of interest from a dynamic system using indirect and inaccurate sensor observations. For instance, in radar tracking, the aim is often to determine the position and velocity of a moving or stationary aircraft or ship. In communication systems, the concern is to determine the messages transmitted through a noisy channel. In driver assistance systems, the objective is to monitor several features about the driver, the vehicle and the surroundings. There are also several other applications such as forecasting weather or financial trends, predicting house prices, handwriting recognition, speaker identification, and positioning in navigation systems.

The sequential estimation problem can be categorized into three different problem formulations: prediction, filtering and smoothing. The prediction problem is to forecast the values of the parameters of interest, given information up to an earlier time, whereas the filtering problem is about estimating the parameter at the current time, given information up to and including that time. The smoothing problem is to estimate the past state of the parameter using all the observations made. An example from [1] can be used to explain these different problem formulations, in layman terms. Assume that we have received a garbled telegram and that the task is to read it word-by-word and make sense of what the telegram means. The filtering formulation would be to read each word and understand the meaning so far. The prediction formulation would be to guess the coming words, based on what have been read thus far. In the smoothing formulation, the reader is allowed to look ahead one or more words. Clearly, as the idiom quoted in the book goes “it is easy to be wise after the event”, the smoothing formulation will give the best result on average, given that a delay can be tolerated.

In many of the above-mentioned applications, the aim is not only to

estimate the parameters of interest termed ‘states’, but also to describe the uncertainties in the states. The uncertainties are used to describe the reliability or trustworthiness of the produced estimates. Mathematically, an estimate and its associated uncertainty is quantified using either a probability density function (for continuous states) or a probability mass function (for discrete states). In sequential estimation, the probability function of the state, which has the information about its estimate and the corresponding uncertainty, are propagated across time to estimate the subsequent states. For continuous states, one of the most commonly used density functions is the Gaussian density function, which is often referred to as the ‘bell-shaped’ curve. The famous Kalman filter [2] is developed as a solution to the filtering problem when the uncertainties are modelled using Gaussian density functions. There also exist (analytical) solutions to the smoothing problem with Gaussian densities.

Even though the Gaussian density models and the Kalman filter solutions work well for a wide range of applications, this may not be enough for complex systems. There are many applications where the uncertainty in the evolution of the state or the observation noise cannot be accurately modelled using Gaussian densities. For instance, in the data association problem, observations are often received from objects that are not of interest and the information regarding which measurement belongs to the target of interest is not available. In such cases, the uncertainty about the states are clustered in several small regions where each region corresponds to each measurement. This happens in ship surveillance, when false measurements are received from reflections of the sea, and in air traffic surveillance, where extraneous observations from clouds and birds are received. In these kinds of scenarios, instead of a single Gaussian density, the system or the observations are often modelled using what is called a Gaussian-mixture density. In essence, the uncertainty of the state can be described using a Gaussian mixture where we have a Gaussian component for each cluster/region around which the uncertainty/data is centered, along with a weight that captures the intensity. The advantage of using a Gaussian mixture (GM) is that it is made up of several Gaussian components, which allows one to extend the Kalman filter solutions to these problems as well. However, in most problems, the number of possibilities and thus the number of Gaussian components in the mixture grows with time, which adds to the complexity of the algorithms.

In the data association problem, the interpretation of a GM density of the state is that we have Gaussian uncertainty about the state for every

possibility of matching the objects (also referred to as targets) of interest to the individual observations from the sensors. To give a sense of the number of possibilities, assume that there are k targets and n measurements. Then, the number is $\frac{n!}{(n-k)!}$ where $n! = n \times (n-1) \times \dots \times 1$, assuming that every target produces a measurement at each time instant. Thus, at each time instant, we have a GM with a large number of components. When this density is propagated to the next time to perform the Bayesian inference we discussed in the beginning of the chapter, the complexity of the problem explodes. Even for single target, i.e. for $k = 1$, the computation of the optimal solution becomes intractable. Thus, approximations are inevitable. In this thesis, we provide efficient and effective solutions for both single and multi-target scenarios.

Another aspect considered in this thesis is the performance evaluation of trajectory estimation algorithms. The main objective of this part is to be able to quantify the similarity between the ground truth and the estimates returned by an algorithm. We might observe that there is a good match between some of the states in the ground truth and the estimates. We will want to quantify the similarity to judge how different algorithms perform. Besides this kind of error, it is possible that there are certain states in the ground truth that do not have any good match in the estimates, or vice versa. We would like to take into account these kinds of errors as well when defining the similarity between the ground truth and the estimates. In this thesis, we have focussed on mathematically quantifying these kind of similarities for trajectory estimation.

The research presented in this thesis has been sponsored by Vinnova (The Swedish Governmental Agency for Innovation Systems), under the research program "Nationella flygtekniska forskningsprogrammet" (NFFP 6), and by Electronic Defence Systems, Saab AB.

Outline of the thesis

The thesis is divided into two parts, where Part I presents the theoretical background of the Gaussian mixture smoothing problem and Part II contains a set of appended papers. Part I contains six chapters, among which Chapter 2 presents the mathematical formulations of the filtering and smoothing problems and different models used. Chapter 3 discusses in detail the Gaussian mixture smoothing problem in the context of single trajectory estimation, and the difficulties involved. In Chapter 4, we discuss the data association problem in multi target tracking. Chapter 5 presents

a summary of the metric problem. In Chapter 6, we provide a summary of the contributions in the appended papers and also discuss possible future research directions.

In Part II, the contributions of the thesis are presented in Papers I through VI. In Paper I and Paper II, we considered the data association problem in estimating a single target trajectory in the presence of clutter. Typically, this is carried out using pruning approximations on GMs. In the papers, we present an in-depth study of how to design forward-backward smoothing and two-filter smoothing for Gaussian mixtures, based on both merging and pruning approximations. In Paper III and Paper IV, we consider the data association problem when estimating multiple target trajectories in the presence of clutter. We present two solutions based on expectation maximization (EM) that are iterative. The major consequence of applying EM is that the problem of GM smoothing reduces to Gaussian smoothing in each iteration. In Paper V and Paper VI, we address the problem of defining metrics for sets of target states and for sets of trajectories, respectively.

Chapter 2

Models, objectives and conceptual solution

In trajectory estimation, the goal is to sequentially estimate an unknown variable, given noisy observations of the variable. According to the Bayesian estimation principle, which is commonly used for these problems, the idea is that based on our prior knowledge of the process, we predict the variable with some uncertainty. Then, at a point when new information is available, the prediction is updated to get the ‘posterior’. For trajectory estimation, the term ‘posterior’ takes different meanings depending on the set of measurement that we condition on. When the most recent state of the trajectory is newer than the complete collection of measurements over time, we have a prediction problem; for equally new information we have a filtering problem; and, for a state that is older than the newest piece of information, we have a smoothing problem.

In this chapter, we briefly discuss the mathematical representation of the variables and how the trajectory estimation problem can be posed. We also present a brief summary of the models used in trajectory estimation.

2.1 State space representation

In a state-space representation, the unknown variable to be estimated is termed the ‘state’. The state variable at time k is here denoted as $x_k \in \mathbb{R}^n$ and the observed data as $z_k \in \mathbb{R}^m$. The time variability of the state is described by a motion model while the relation between the state and the measurements are given by a sensor model. The implicit Markovian assumption of the state space is that the state x_k at time k , given all the states until time $k - 1$, depends only on the state x_{k-1} at time $k - 1$. The

motion model can then be written as

$$x_k = g_k(x_{k-1}, v_k), \quad (2.1)$$

where v_k is the process noise. Using this motion model one can describe the transition model $f_k(x_k|x_{k-1})$. It is assumed that we have some knowledge of the state at time 0, defined by the prior density $p_0(x_0)$.

The measurement z_k is given by the sensor model

$$z_k = h_k(x_k, w_k), \quad (2.2)$$

where w_k is the measurement noise random variable. The sensor model is used to obtain the likelihood function $p_k(z_k|x_k)$. In the remainder of the introductory chapters, the subscript k in the notation of the functions $g_k(\cdot)$, $h_k(\cdot)$, and $p_k(\cdot|\cdot)$ will be dropped without loss of generality and for ease of writing, and be represented as $g(\cdot)$, $h(\cdot)$, and $p(\cdot|\cdot)$.

2.2 Problem statement and conceptual solution

The posterior density of the state x_k is used to determine our estimates of the state and also describes our uncertainties in the state. The objective is to recursively compute the posterior density of the state vector x_k using the Bayesian principle [1]. In the prediction problem, the goal is to obtain the density $p(x_k|z_{1:l})$ for $l < k$, given the measurements obtained from time 1 to time l , denoted $z_{1:l}$. In the filtering problem, the goal is to obtain the posterior density $p(x_k|z_{1:k})$ of the state x_k . In the smoothing problem, we are interested in computing the posterior density $p(x_k|z_{1:K})$, where $K > k$. Below, we present the conceptual solutions to these problems.

2.2.1 Prediction and filtering

The prediction and filtering densities can be obtained recursively in two steps, namely, prediction and update, using the prior $p(x_0)$, the process model $f(x_k|x_{k-1})$ and the likelihood $p(z_k|x_k)$. The one-step prediction (where $l = k - 1$ in $p(x_k|z_{1:l})$) gives the prediction density at time k by evaluating the integral

$$p(x_k|z_{1:k-1}) = \int p(x_{k-1}|z_{1:k-1})f(x_k|x_{k-1})dx_{k-1}, \quad (2.3)$$

2.2. PROBLEM STATEMENT AND CONCEPTUAL SOLUTION

where $p(x_{k-1}|z_{1:k-1})$ denotes the filtered density at time $k-1$, and $p(x_k|z_{1:k-1})$ the prediction density at time k . An update of the prediction density at time k gives the filtered density at time k as

$$p(x_k|z_{1:k}) \propto p(x_k|z_{1:k-1})p(z_k|x_k). \quad (2.4)$$

The constant of proportionality in the above equation is $\frac{1}{p(z_k|z_{1:k-1})}$, where

$$p(z_k|z_{1:k-1}) = \int p(x_k|z_{1:k-1})p(z_k|x_k)dx_k. \quad (2.5)$$

It should be mentioned here that the equations in (2.3), (2.4) and (2.5) provide the theoretical solutions but, in practice, these equations are in general not tractable. For instance, the integrals cannot be computed accurately, or the representation of the different densities in these equations can be intractable and so on.

2.2.2 Smoothing

Similar to the filtering problem, sequential estimation of the smoothed posterior can be obtained using the Bayesian principle. Though the approaches discussed here have been designed towards fixed-interval smoothing, they are in their contextual form, applicable to the fixed-lag and fixed point smoothing as well [1]. One can also refer to [3] for accumulated state density formulation of the smoothing problem.

The first approach is forward–backward smoothing (FBS) [4]. As the name suggests, the first step is forward filtering from time 1 to K , to obtain the filtered density $p(x_k|z_{1:k})$ at each k . This is followed by backward smoothing from time K to time 1. The backward smoothing step at time k uses the smoothed density at time $k+1$ together with the filtering densities at time k as

$$p(x_k|z_{1:K}) = p(x_k|z_{1:k}) \int \frac{p(x_{k+1}|z_{1:K})}{p(x_{k+1}|z_{1:k})} f(x_{k+1}|x_k) dx_{k+1}. \quad (2.6)$$

The integral in the above equation is proportional to $p(z_{k+1:K}|x_k)$, termed as the backward likelihood in this thesis. Therefore, it is possible to interpret the division in the backward smoother as computing the backward likelihood implicitly.

The second approach to smoothing is the two-filter smoothing (TFS) method [5]. To obtain the smoothed density at time k by this method, forward filtering is performed from time 1 to k to get the filtered density

$p(x_k|z_{1:k})$ and backward filtering is run from time K to time k to get the backward likelihood $p(z_{k+1:K}|x_k)$. The product of the two filter outputs gives the smoothed density,

$$p(x_k|z_{1:K}) \propto p(x_k|z_{1:k})p(z_{k+1:K}|x_k). \quad (2.7)$$

The backward filtering, similar to the forward filter, is performed recursively using two steps: update and retrodiction. The update step computes the likelihood

$$p(z_{k+1:K}|x_{k+1}) = p(z_{k+1}|x_{k+1})p(z_{k+2:K}|x_{k+1}) \quad (2.8)$$

and the retrodiction step computes the backward likelihood as

$$p(z_{k+1:K}|x_k) = \int p(z_{k+1:K}|x_{k+1})f(x_{k+1}|x_k)dx_{k+1}. \quad (2.9)$$

Comparing the individual terms in (2.6) and (2.7) and using (2.9), one can observe that the difference between the two smoothing methods arises due to the difference in the ways the term $p(z_{k+1:K}|x_{k+1})$ is computed. FBS computes it from the division of the prediction and smoothing densities

$$p(z_{k+1:K}|x_{k+1}) \propto \frac{p(x_{k+1}|z_{1:K})}{p(x_{k+1}|z_{1:k})}. \quad (2.10)$$

whereas the TFS method uses a filtering approach as in (2.8).

2.3 Models

In this section, we discuss the different models relevance to trajectory estimation. We present them in different categories based on their properties. All these models determine whether or not the integrals in the conceptual solutions presented in the last section are tractable or not. For instance, as we discuss in the next chapter, when we have a single target with linear process and measurement models, along with Gaussian noise terms and Gaussian prior densities, the prediction, filtered and smoothed densities are all Gaussian densities. In this case, the Kalman filter and the Rauch-Tung-Striebel (RTS) smoother provide a recursive solution to obtain the mean and the covariance of these densities in closed form. Below we summarize some of the possible models used in the trajectory estimation literature.

The process model $g(x_{k-1}, v_k)$ and/or the measurement model $h(x_k, w_k)$ can be non-linear functions of the state and the noise variables. A commonly used non-linear measurement model is when we have range and bearing measurements and the state that we are interested in is Cartesian position.

2.3. MODELS

These kind of non-linear functions are often handled using extended Kalman filter (EKF) [1, 6], unscented Kalman filter (UKF) [7, 8], quadrature Kalman filter [9], cubature Kalman filter (CKF) [10] or the particle filter [11–16].

Often in applications, we receive measurements, not only from the target of interest, but also from sources that are not of interest to us. These measurements are termed clutter and often yield uncertainties in the data associations. The problem arises when it is not immediate which measurements are from the single target and which are from clutter. The data association problem also arises when we have multiple targets in the region of interest. Again, we receive a set of measurements which may not have the target identity. The problem is only aggravated when we also have the clutter data in addition.

Chapter 3

Single trajectory estimation

In this chapter, we briefly discuss the various challenges in filtering and smoothing problems that arise in the single-trajectory estimation problem. We discuss in detail the Gaussian mixture filtering and smoothing problems, which are focus areas of the thesis.

3.1 Linear and Gaussian filtering and smoothing

Assume that the prior, $p(x_0)$, is a Gaussian density, and that the motion and sensor models are linear functions of the state vector x_k , with additive Gaussian noise, i.e.,

$$x_k = F_k x_{k-1} + v_k \tag{3.1}$$

and

$$z_k = H_k x_k + w_k, \tag{3.2}$$

where $F_k \in \mathbb{R}^{n \times n}$, $H_k \in \mathbb{R}^{m \times n}$, $v_k \sim \mathcal{N}(0, Q_k)$ and $w_k \sim \mathcal{N}(0, R_k)$. Then, it can be shown that the posterior densities are Gaussian and have closed form expressions [2]. Again, for convenience of writing, the subscript k will be dropped from the matrix notations. In this section, we discuss the algorithms to obtain the mean and covariance of the filtered and smoothed densities.

3.1.1 Kalman filtering

Let the prediction density, $p(x_k|Z_{1:k-1})$, and filtered density, $p(x_k|Z_{1:k})$, be denoted as $\mathcal{N}(x_k; \mu_{k|k-1}, P_{k|k-1})$ and $\mathcal{N}(x_k; \mu_{k|k}, P_{k|k})$, respectively. The notation $\mathcal{N}(x; \mu, P)$ denotes a Gaussian density in the variable x with mean μ and covariance P . The goal of prediction and filtering is then to find the

first two moments of the corresponding Gaussian densities. The ubiquitous Kalman filter equations [2] provide closed-form expressions for the first two moments of the prediction and filtered densities in (2.3) and (2.4). The prediction equations are given by

$$\mu_{k|k-1} = F\mu_{k-1|k-1} \quad (3.3)$$

$$P_{k|k-1} = FP_{k-1|k-1}F^T + Q \quad (3.4)$$

and the update equations by

$$S_k = HP_{k|k-1}H^T + R \quad (3.5)$$

$$K_k = P_{k|k-1}H^T S_k^{-1} \quad (3.6)$$

$$\tilde{z}_k = z_k - H\mu_{k|k-1} \quad (3.7)$$

$$\mu_{k|k} = \mu_{k|k-1} + K_k\tilde{z}_k \quad (3.8)$$

$$P_{k|k} = (I_n - K_kH)P_{k|k-1}. \quad (3.9)$$

where I_n is an n -by- n identity matrix. The so-called innovation \tilde{z}_k , and S_k , the innovation covariance, describe the expected measurement distributions. K_k is the Kalman gain, which can be viewed as the weight for new information in the innovation compared to the prediction.

3.1.2 Smoothing

Below we provide the two versions—forward-backward smoothing (FBS) and two-filter smoothing (TFS)—of the smoothing algorithm discussed in Section 2.2.2 to obtain the mean and covariance of the smoothed density under linear and Gaussian assumptions.

For the FBS method, the Rauch-Tung-Striebel (RTS) [4] smoother gives the closed-form expressions for the mean and covariance of the smoothed density. Using notations similar to the ones for the prediction and filtered densities, the smoothed density at time k is denoted as $\mathcal{N}(x_k; \mu_{k|K}, P_{k|K})$. The RTS equations are

$$\mu_{k|K} = \mu_{k|k} + C_k (\mu_{k+1|K} - \mu_{k+1|k}) \quad (3.10)$$

$$P_{k|K} = P_{k|k} + C_k (P_{k+1|K} - P_{k+1|k}) C_k^T, \quad (3.11)$$

where

$$C_k = P_{k|k}F^T P_{k+1|k}^{-1} \quad (3.12)$$

is similar to the Kalman gain in the Kalman filter equations (3.3) to (3.9).

3.2. NON-LINEAR MODELS

For the TFS method, the work in [17] provides the closed-form solution for the moments of the smoothed density. Let the likelihoods be denoted as $p(Z_{k+1:K}|x_{k+1}) = \mathcal{N}(U_{k+1}x_{k+1}; \psi_{k+1}, G_{k+1})$ and $p(Z_{k+1:K}|x_k) = \mathcal{N}(J_k x_k; \eta_k, B_k)$. Let the starting conditions at time K be $J_K = [\]$, $\eta_K = [\]$ and $B_K = [\]$. The update step in (2.8) of the backward filter is then given by

$$U_{k+1} = \begin{bmatrix} J_{k+1} \\ H \end{bmatrix} \quad (3.13)$$

$$\psi_{k+1} = \begin{bmatrix} \eta_{k+1} \\ z_{k+1} \end{bmatrix} \quad (3.14)$$

$$G_{k+1} = \begin{bmatrix} B_{k+1} & \mathbf{0} \\ \mathbf{0} & R \end{bmatrix} \quad (3.15)$$

while the retrodiction step in (2.9) is given by

$$J_k = U_{k+1}F \quad (3.16)$$

$$\eta_k = \psi_{k+1} \quad (3.17)$$

$$B_k = U_{k+1}QU_{k+1}^T + G_{k+1}. \quad (3.18)$$

Using the outputs of the forward filter and the backward filter at time k , the smoothed density in (2.7) is given by

$$\mu_{k|K} = \mu_{k|k} + W_k (\eta_k - J_k \mu_{k|k}) \quad (3.19)$$

$$P_{k|K} = (I_n - W_k J_k) P_{k|k}, \quad (3.20)$$

with gain

$$W_k = P_{k|k} J_k^T (J_k P_{k|k} J_k^T + B_k)^{-1}. \quad (3.21)$$

Note that the above three equations have similarities to the Kalman update equations in (3.8) and (3.9). Here, the filtering parameters, $\mu_{k|k}$ and $P_{k|k}$, are updated with the innovation from the future measurements from time $k+1$ to K , whereas in the Kalman filter the prediction parameters are updated with the innovation from the measurement at time k .

3.2 Non-linear models

When the motion model $g(\cdot)$ and/or the measurement model $h(\cdot)$ are non-linear or when the noise is not additive Gaussian, the posterior density for the prediction, filtering and smoothing is, in general, not a Gaussian density. One example is when we obtain range and bearing measurements

from a radar and want to track the position and the velocity of the target. The optimal solution then becomes intractable as the integrals cannot be computed in closed form. Approximations and sub-optimal approaches are therefore inevitable. There are several sub-optimal approaches to estimate the filtered density in this case, some of which are discussed in this section such as the Gaussian and particle filters. We discuss Gaussian mixtures, which is applicable when the posterior has multimodal shape, in more detail in subsequent sections of this chapter.

The smoothing problem has additional challenges compared to the filtering problem. First, the equation in the FBS method involves division of densities, which is difficult to compute for arbitrary densities. Second, the accuracy of the approximations in the forward filtering highly affects the backward smoothing and the smoothed density. The TFS method, on the other hand, does not involve density divisions and the two filters can ideally be run independently of each other. However, the likelihood $p(z_{k+1:K}|x_k)$ is not, in general, a normalizable density function, which limits the possibilities to apply conventional approximation techniques for densities during the backward filtering. Due to these additional complications, applying the techniques used for non-linear filtering to non-linear smoothing does not always produce fruitful results. In this section, we also discuss the challenges in extending techniques such as sequential Monte Carlo sampling, linearization and sigma-point methods to the smoothing problem.

3.2.1 Gaussian filters

One approach to handle non-linear models is to approximate the posterior density as a Gaussian density. The methods that use this approach are called Gaussian filters/smoothers, named appropriately. There are several methods to make a Gaussian approximation of the filtered density and to compute its first two moments. One method is based on linearizations of the functions, $g(\cdot)$ and $h(\cdot)$, after which the Kalman filter equations in (3.3) to (3.9) can be used to obtain the mean and covariance of the Gaussian approximation of the filtered density. The famous extended Kalman filter [1, 6, 18] and the many variants of it are based on this approach. Though these algorithms work for a good number of models, their performance deteriorates when the functions are highly non-linear.

Another type of methods used to obtain a Gaussian approximation of the filtered density is based on sigma-points, such as the unscented Kalman filter [7, 8], the quadrature Kalman filter [9] and the cubature Kalman fil-

3.2. NON-LINEAR MODELS

ter [10]. In these methods, a handful of points, termed sigma-points, are chosen deterministically based on the first two moments of the prior density. The sigma points are then propagated through the non-linear models to obtain the means and covariances used to compute the moments of the Gaussian approximation of the filtered density. The sigma-point methods also implicitly perform a linearisation using statistical linear regression [19].

The analogue of Gaussian filtering methods, such as the extended Kalman filter and the unscented Kalman filter, exists for TFS of non-linear models. The extended Kalman smoother [20], similar to its filtering counterpart, has poor performance when the non-linearity is severe. The unscented Kalman smoother [21], [20, Chap. 7] needs the inverse of the dynamic model functions, which may not be feasible in all scenarios. The unscented RTS smoother [22] is the FBS version of a Gaussian smoother and is shown to have similar performance as the unscented Kalman smoother, but without the need of inverting the model functions.

3.2.2 Particle filters

Particle filters or sequential Monte Carlo filters [11–16] are based on representing the density $p(x)$ with a set of randomly drawn samples $x^{(m)}$, termed ‘particles’, along with their corresponding weights. The particles define the positions of Dirac delta functions such that the weighted sum of the Dirac delta functions of the particles provides a good approximation of the true density:

$$p(x) \approx \frac{1}{N} \sum_{m=1}^N \delta(x - x^{(m)}). \quad (3.22)$$

These methods use the concept of importance sampling, where the particles are generated from a proposal density, which is simpler to generate the samples from, instead of the true density. The particles are propagated through the process model and the weights are updated using the likelihood to obtain the posterior density. The choice of proposal density is crucial to particle filters, and the proposal density should have the same support as the true density and should be as similar to the true density as possible.

The advantages of particle filters are that the performance of such filters is unaffected by the severity of the non-linearity in $g(\cdot)$ and $h(\cdot)$, that they are asymptotically optimal also when the functions are non-linear, and that they are often easy to implement. However, they can be computationally demanding as the dimension of the state vector increases. Another problem with particle filters is that they degenerate, which means that

the weights of most particles become zero. This can be overcome by resampling [16] frequently, where multiple copies of the ‘good’ particles with significant weights are retained and the ‘poor’ particles are removed.

Similar to the filtering method, sequential Monte Carlo smoothing is based on approximating the smoothed posterior density using a set of particles. In particle Markov chain Monte Carlo (MCMC) methods [23], particle filters are used to approximate the joint posterior distribution, which is then used to generate proposals for MCMC. In case of FBS based on these methods, a vanilla version works well when $k \approx K$, where K is the batch duration. However, when $k \ll K$ [24], because of successive resampling steps, the marginal density becomes approximated by a single particle which leads to deteriorated performance. This is the degeneracy problem that is inherent in particle filters [11]. One simple approach is to use the forgetting properties of the Markov model, i.e., to approximate the fixed-interval smoothed density $p(x_k|z_{1:K})$ using the fixed-lag smoothed density $p(x_k|z_{1:k+\delta})$ [25, 26]. Unfortunately, automatic selection of δ is difficult.

In case of TFS, it is not straightforward to approximate the output of the backward filter using particles, as it is not a normalizable density. The artificial prior method [5] uses the auxiliary probability density $\tilde{p}(x_k|z_{k+1:K})$ instead of the likelihood $p(z_{k+1:K}|x_k)$. The auxiliary density is obtained using what is called artificial prior densities. The choice of the artificial prior plays a major role in the performance of the TFS algorithm for particle methods.

3.3 Gaussian mixture filtering and smoothing

There are many applications in which we receive several measurements on the state variable, where the reliability of the measurements can vary. The likelihood in these applications are conveniently modelled as mixtures. In the classic data association problem, where we do not have information about which measurement corresponds to which target upon receiving a set of measurements, the posterior density is a Gaussian mixture, even if we assume the motion and measurements models are linear and Gaussian. We discuss more on this problem in the next chapter.

Gaussian mixtures are weighted sum of Gaussian densities, which usually make a good approximation for the multi-modal densities. In this section, we explain that when the likelihood and/or the state transition density are Gaussian mixtures, the true posterior densities after filtering and smoothing

are also Gaussian mixtures. The number of terms in the GM usually grows exponentially with time, and we therefore need to constrain the number of terms. In these situations, reduction algorithms can be used to approximate the posteriors. In this section, we provide a brief overview of the most commonly used mixture reduction methods and discuss the challenges in applying these to smoothing problems.

3.3.1 Optimal solution

It was presented in the last chapter that the forward-backward smoothing (FBS) method is based on forward filtering and backward smoothing while the two-filter smoothing (TFS) method involves forward filtering and backward filtering. These steps involve the prediction, update and retrodiction steps stated in equations (2.3), (2.4), (2.8) and (2.9). One can notice that all these equations involve products of functions, which in this case are Gaussian mixtures. When the state transition densities and the likelihoods are both GMs, one can use the fact that a product of GMs yields a GM and show that the posterior densities are all Gaussian mixtures. The number of components in the resulting GM is the product of the number of components in the individual mixtures, which explains why the number of components grows exponentially with time.

Forward filtering

One can show that, starting with a GM prior, the prediction and the update steps of forward filtering result in a Gaussian mixture posterior density. Evaluating these steps with GMs is equivalent to using Kalman filters, one for every triplet of Gaussian components in the prior $p(x_{k-1}|z_{1:k-1})$, the transition density $f(x_k|x_{k-1})$ and the likelihood $p(z_k|x_k)$, yielding a Gaussian term in the posterior $p(x_k|z_{1:k})$. The term in the constant of proportionality $p(z_k|z_{1:k-1})$ in (2.5), is not calculated explicitly in the update step of the Kalman filter, which involves product of Gaussian densities. However, in the case with GMs, this constant of proportionality is used in the updated weight calculation. The updated weight for the resulting Gaussian component is given by the product of the individual weights of the components in the prediction density and the likelihood along with the constant of proportionality.

Backward smoothing of FBS

The backward smoothing of FBS involves a division of the smoothed and the prediction GM densities as in equation (2.6). Starting from time K ,

using the principle of mathematical induction, it can be shown that the division results in a GM and therefore the smoothed posterior $p(x_k|z_{1:K})$ is also a GM which has the same number of components for $k = 1, \dots, K$. The weights of the components in the the smoothed density at time k are the same as the weights of the components in the smoothed density at time $k + 1$. Instead of performing the division, an equivalent way of obtaining the smoothed posterior is as follows [3, Sec. V A]: starting from $k = K - 1$, the RTS recursions are used for every triplet of associated components in the smoothed density at time $k + 1$, the prediction at time k and in the filtered density at time k , to compute the smoothed density at time k .

Backward filter of TFS

In the backward filter of TFS, we need to compute the backward likelihood as in equations (2.8) and (2.9). The ideas in forward filtering cannot be applied directly to the backward filter because often the likelihoods can be of the form

$$w_0 + \sum_i w_i \mathcal{N}(H_i x_k; \mu_i, P_i) \quad (3.23)$$

where different H_i can capture different features of the state x_k . Strictly speaking, these are not Gaussian mixture densities; they are neither Gaussian nor densities in x_k . We refer to them as reduced dimension Gaussian mixtures in this thesis. To compute the product of likelihoods, one can use the following general product rule:

$$w_i \mathcal{N}(H_i x; \mu_i, P_i) \times w_j \mathcal{N}(H_j x; \mu_j, P_j) = w_{ij} \mathcal{N}(H_{ij} x; \mu_{ij}, P_{ij}) \quad (3.24)$$

where

$$w_{ij} = w_i w_j \quad (3.25)$$

$$H_{ij} = \begin{bmatrix} H_i \\ H_j \end{bmatrix} \quad (3.26)$$

$$\mu_{ij} = \begin{bmatrix} \mu_i \\ \mu_j \end{bmatrix} \quad (3.27)$$

$$P_{ij} = \begin{bmatrix} P_i & 0 \\ 0 & P_j \end{bmatrix}. \quad (3.28)$$

Using this in equations (2.8) and (2.9), one can show that the output of the backward filter has a structure similar to the inputs as in (3.23). The smoothed density is given by the product of the outputs of the two filters, that can be computed similarly to the update step in the forward filter of GMs, including the weight update using the proportionality constant as discussed in Section (3.3.1).

3.4 Gaussian mixture reduction

The number of components in the resulting GM, after update, prediction and retrodiction iterations, grows exponentially with time. Therefore, approximations are necessary to reduce the number of components. There are several Gaussian mixture reduction (GMR) algorithms that are well studied in the literature, which can be used for filtering and smoothing. The GMR algorithms are based on pruning insignificant components from the GM and/or merging similar components.

3.4.1 Pruning

The number of components in the posterior GM can be prevented from growing exponentially by pruning some of the components after each iteration. There are several pruning strategies that can be adopted. Three methods that are commonly used are threshold-based pruning [27], M -best pruning [28–30] and N -scan pruning. In threshold-based pruning, only the components that have a weight greater than a predefined threshold are retained and used for prediction in the next iteration. The number of components in the resulting GM can vary based on the threshold. The idea behind the M -best pruning algorithm is that only the nodes with the M highest weights (or association probabilities) are retained.

To explain the N -scan pruning [27], which is designed for reduction during filtering, let us say we are interested in performing pruning at time k . We pick the component that has the maximum weight. Starting from this component, we trace backwards N steps to find its parent component, at time $k - N$. Only the offspring at time k , of this parent node at time $k - N$, are retained. To be mentioned here is that the multiple hypothesis tracking (MHT) filtering [27] is often based on N -scan pruning.

3.4.2 Merging

One can also use merging of similar components to reduce the number of components in a GM. There are several merging algorithms such as Salmond's [31–33], Runnalls' [34], Williams' [35] algorithms and many more [36–39]. These algorithms work based on the following three steps:

1. Find the most suitable pair of components to merge according to a 'merging cost' criterion.
2. Merge the similar pair and replace the pair with the merged Gaussian component.

3. Check if a stopping criterion is met. Otherwise, set the reduced mixture to the new mixture and go to step 1.

The merging cost in step 1 looks for similarity of the components and it can be different across algorithms. A few of the commonly used merging costs are the Kullback-Leibler divergence [40] and the integral-squared error [41–43]. The merging of the components in step 2 is usually based on moment matching [40]. That is, the moments of the GM before and after merging are the same. The stopping criterion can also vary across algorithms, e.g., it can be based on if the components in the reduced mixture is at a manageable number. In certain algorithms, it is checked based on that the components in the reduced GM are not similar.

3.4.3 Choice of GMR

Two main criteria in choosing the appropriate GMR algorithm are the computational complexity involved and the accuracy. Most of the pruning algorithms are usually simpler to implement, compared to merging. There is information about the uncertainty of the estimate in the covariance matrices of the pruned components. So, as a result of pruning, we might have underestimated uncertainties. In contrast, for merging, the uncertainty is preserved because of moment-matching. However, the merging algorithms are more computationally intensive than pruning. As a trade-off between complexity and accuracy of the uncertainty, it may be more feasible to use a combination of pruning and merging. Pruning ensures that the components with negligible weights are removed, without being aggressive. Merging reduces the number of components further, but keeping the moments of the retained density the same as before.

3.4.4 GMR for FBS and TFS

Applying GMR, both pruning and merging for the forward filtering is straightforward. When the forward filtering is based on pruning, it is trivial to perform the backward smoothing of the FBS similar to the optimal solution, using the filtered densities. Starting from the last time instant, RTS is performed backwards on the individual retained components. This method suffers from degeneracy similar to particle smoothing. This is because for $k \ll K$, the number of components in the forward filter that corresponds to the components in the smoothed posterior will be one. A solution to the degeneracy is to perform FBS based on merging, something that has not been explored much in the literature. The main challenge is that for the backward smoothing, the associations across components are no longer

3.4. GAUSSIAN MIXTURE REDUCTION

simple, to use RTS directly and compute the weights of the smoothed density. In Paper I [44] of this thesis, the problem of FBS based on merging is investigated.

The literature on TFS for Gaussian mixture densities is also sparse. The two filters of the TFS can be run independently of each other. This allows the GMR algorithms to be used on both the filters. Then the difficulty is in using the Gaussian mixture reduction techniques in the backward filter, since its output is not a density function. So, the GMR algorithms discussed here cannot be applied directly. In Paper II [45] of this thesis, we propose a method called smoothed posterior pruning, through which pruning can be employed in the backward filter.

Chapter 4

Multi-trajectory estimation

In the previous chapter, we presented a brief discussion of the data association problem. Even with linear Gaussian assumptions, the number of Gaussian terms in the Gaussian mixture form of the posterior density grows exponentially. In this chapter, we discuss the problem further in the multi-target setting. Additional challenges are posed by having a set of data from multiple targets, where the target identities are not available.

4.1 Data association

In the presence of multiple point targets, each of which follows a linear-Gaussian process and measurement model, at each time we observe a set of measurements. If the identity of a target is not available in the measurements, then there is uncertainty about which measurement belongs to which target. In case of a single trajectory, each hypothesis is an event of assigning a target to a measurement from the set. In this case, the number of possibilities at each time is the same as the number of measurements, which when multiplied across time becomes exponential.

With multiple targets, at each time, the problem is even worse. Let us say we have n targets and m measurements at a particular time. Now, each hypothesis, often referred to as a global hypothesis, is the event of associating the n targets to the m measurements (assuming $n \leq m$) such that a target is assigned to at most one measurement and a measurement is assigned to at most one target. The number of possibilities is combinatorial given by $\frac{m!}{(m-n)!}$; for instance, if we assume $n = 2$ and $m = 3$, the number of combinatorial possibilities is 6; if we double them up, $n = 4$ and $m = 6$, the number of possibilities is 360. If we also consider the possibility that a target does not need to generate a measurement, in other words, a target can

be missed, the number of possibilities is even higher, and again combinatorial. Across time, the complexity of the problem gets multiplied. Therefore, the optimal way of solving the problem considering all the possibilities is intractable. Below we discuss briefly the traditional approach taken.

Let K denote the batch duration, k the time index, N_K the number of targets in the entire batch duration and M_k the number of measurements obtained at k . Assume the state variable is $X = (X_{k,i} : k = 1, \dots, K, i = 1 \dots, N_K)$ and the measurement is $Z = (Z_{k,j} : k = 1, \dots, K, j = 1 \dots, M_k)$ where i stands for the target index and j for the measurement index.

If we know the posterior density $p(X|Z)$, we can estimate the states, which are tractable and straightforward if there is no uncertainty in the measurement origin. However, in the multi-target tracking problems, the measurement set comprises the measurements from the targets that are detected as well as the clutter measurements, and the origin of the measurements in the set are not known. To handle this uncertainty in the measurement origin, one traditional way is to introduce a data association variable $\phi = \{\phi_{k,i}, \forall k, i\}$, where $\phi_{k,i} = j$ denotes the assignment of the target i at time k to the measurement $Z_{k,j}$. With these variables, the density of interest becomes $p(X, \phi|Z)$, using which the estimates can be computed. Though the introduction of this variable makes it easier to represent the measurement uncertainty, the estimation problem is still intractable due to the sheer number of possibilities of ϕ . For instance, consider $\hat{X}_{\text{MAP}} = \arg \max_X p(X|Z) = \arg \max_X \max_{\phi} p(X, \phi|Z)$. The optimal point is searched over all the possibilities of ϕ , which is exponential in the number of measurements. Thus, sub-optimal approaches are inevitable. In the remainder of this section, we give a brief overview of some of the existing sub-optimal algorithms to estimate X . Adhering to the conventional terminology, we refer to an instance of ϕ as the data association hypothesis.

4.2 Tracking algorithms

The existing algorithms for tracking can be broadly categorised into two: the ones that jointly estimate X and ϕ , and the others that estimate X while marginalizing ϕ . Global nearest neighbour (GNN) [27] and multiple hypothesis tracking (MHT) [27, 46, 47] belong to the first category, whereas joint probabilistic data association (JPDA) [27, 48–55], probabilistic MHT (PMHT) [56–59] and their variants belong to the second one. There are also sampling-based algorithms like Markov chain Monte Carlo data association (MCMCDA) [60]. In this algorithm, an estimate of a hypothesis is

4.2. TRACKING ALGORITHMS

obtained by making several random changes to the existing hypothesis. The computational and memory requirements of this algorithm are very high, in general. In this section, we first focus on the first category of algorithms that estimate ϕ , while estimating X , followed by the second category that estimate X .

4.2.1 Global nearest neighbour

In case of the global nearest neighbour algorithm, at each time instant k , the best data association hypothesis is chosen to be the one with the largest $\Pr\{\phi_k|Z_{1:k}\}$, where ϕ_k stands for the data association at time k and $Z_{1:k}$ for the set of measurements from time 1 to k , respectively. This hypothesis is propagated to the next time and associated with the new set of measurements to form a new set of hypotheses. The best hypothesis is chosen to be the one with the largest $\Pr\{\phi_{1:k+1}|Z_{1:k+1}\}$ and the whole procedure is repeated for subsequent time instants. This algorithm is very simple to implement using 2-D assignment algorithms [61] such as the auction algorithm [62] or Jonker-Volgenant-Castanon (JVC) [63], but since it makes hard decisions every time instant, it underestimates the covariance and can often lead to track loss.

4.2.2 Multiple hypothesis tracking

Similar to GNN, MHT generally makes hard decisions when estimating ϕ . However, unlike GNN, the MHT algorithms make hard decisions based on multiple scans of data. The commonly used N -scan pruning based track-oriented MHT algorithm chooses the best set of hypotheses at time k based on the last N scans of data and, propagates them, and repeats the procedure again. In essence, the algorithm estimates the best hypothesis at time $k - N$ based on the information until the current time instant k . The N -scan pruning algorithm is typically implemented using the N -dimensional assignment algorithms as in [64–66]. Another popular version of MHT is to retain the M -best hypotheses at each time and propagate only these M hypotheses to the next time instant. This is typically implemented using Murty’s algorithm [28–30, 67].

As can be noticed, MHT maintains multiple data association hypotheses every time instant and, hence the name ‘multiple hypothesis tracking’. The larger the N (or M) is, the more accurate the estimates are. However, larger N leads to higher complexity; and smaller N leads to the ‘short’ history problem of MHT. That is, the different hypotheses that are maintained each time instant k differ only in the most recent N ($k - (N + 1), \dots, k$)

associations and are identical from time 1 to $k - N$. Therefore, there is only one data association sequence maintained from time 1 to $k - N$ and any new information from future measurements cannot be used to update the data association in those time instants.

4.2.3 Joint probabilistic data association

Let us now shift our focus to the other class of algorithms which estimate X by integrating out ϕ . The JPDA algorithm integrates out ϕ at each time instant and performs moment matching to approximate the distribution over X to a Gaussian density. That is, the possibly multi-modal density $p(X|Z) = \sum_{\phi} p(X, \phi|Z)$ is approximated to a uni-modal density $\tilde{p}(X|Z)$ where the inclusive KL divergence $\text{KL}(p(X|Z)||\tilde{p}(X|Z))$ [68] is minimized. Again, this algorithm is computationally simpler than MHT, but when the approximated posterior density is propagated across time, the performance degrades.

4.2.4 Probabilistic multiple hypothesis tracking

Another popular tracking algorithm is the probabilistic multiple hypothesis tracking (PMHT), which aims to compute the maximum a posteriori (MAP) estimates of the entire sequence of target states. The idea is to obtain these estimates using the expectation maximization (EM) algorithm, where the solution is iterative and involves several local optimisations:

$$\hat{X}^{(n+1)} = \arg \max_X \sum_{\phi} \Pr\{\phi|Z, \hat{X}^{(n)}\} \ln p(X, \phi, Z). \quad (4.1)$$

PMHT allows a target to be associated to multiple measurements, which enables closed-form expressions to compute the marginal association probabilities of the different trajectories. However, this approximation is more suitable for extended target models than the point target model assumptions presented in the beginning of this section. Paper III [69] of this thesis also uses EM similar to PMHT to estimate the states; however, we retain the point-target constraints that a target is assigned to at most one measurement and a measurement is assigned to at most one target.

4.3 EM for data association

Expectation maximization (EM), first discussed in [70, 71], is an iterative technique, widely used to obtain approximate maximum-likelihood estimation (MLE), or MAP estimates of parameters from observed data. In the

4.3. EM FOR DATA ASSOCIATION

EM solution, the model is assumed to have hidden variables that relate the measurements with the states. This makes the posterior density analysis simpler, which is otherwise intractable. In trajectory estimation, we are interested in obtaining estimates of the state vector X . In this section, we propose two versions of EM to obtain the state estimates using the joint density $p(X, \phi, Z)$. To start with, we present a brief introduction to the EM algorithm for MAP estimation.

To derive EM in its general form, we adhere to a notation that is common in the EM literature. According to the notation, θ is the parameter to be estimated, Z the observed data and γ the hidden variable. The MAP estimation of θ is given by,

$$\hat{\theta} = \arg \max_{\theta} p(\theta|Z) = \arg \max_{\theta} \ln \int p(\theta, Z, \gamma) d\gamma. \quad (4.2)$$

Note that the logarithm that has been introduced in the maximization is a monotonically increasing function and does not affect the MAP estimation.

In many applications (including tracking, as will be shown), the integral in the MAP estimation according to (4.2) is not always tractable. To get a tractable approximation, $q_{\gamma}(\gamma)$ over the hidden variable γ is introduced in the objective function in (4.2):

$$\ln p(\theta, Z) = \ln \int q_{\gamma}(\gamma) \frac{p(\theta, Z, \gamma)}{q_{\gamma}(\gamma)} d\gamma \quad (4.3)$$

$$\geq \int q_{\gamma}(\gamma) \ln \frac{p(\theta, Z, \gamma)}{q_{\gamma}(\gamma)} d\gamma \quad (4.4)$$

$$\triangleq \mathcal{F}(q_{\gamma}(\gamma), \theta). \quad (4.5)$$

Jensen's inequality is used to go from (4.3) to (4.4). As can be observed, the term $\mathcal{F}(q_{\gamma}(\gamma), \theta)$ on the right-hand side of (4.4) is a lower bound on the logarithm of the joint density $\ln p(\theta, Z)$ [71] and is a functional of q_{γ} and θ . In EM, this lower bound is increased with iterations such that the difference between the bound and the logarithm of the density is decreased [70]. This is achieved by performing the following set of operations in each iteration ($n + 1$):

$$q_{\gamma}^{(n+1)}(\gamma) = \arg \max_{q_{\gamma}(\gamma)} \mathcal{F}(q_{\gamma}(\gamma), \theta^{(n)}) \quad (4.6)$$

$$\theta^{(n+1)} = \arg \max_{\theta} \mathcal{F}(q_{\gamma}^{(n+1)}(\gamma), \theta). \quad (4.7)$$

The first step is called the the E-step, where the best $q_{\gamma}^{(n+1)}(\gamma)$ is computed given $\theta^{(n)}$. The second step, called the M-step, computes the best θ^n given

$$q_{\gamma}^{(n+1)}(\gamma).$$

In Paper III [69] and Paper IV [72], we have used two approaches in EM to obtain the estimates. In Paper III, we use the EM problem formulation to estimate the state X directly from the joint density $p(X, \phi, Z)$. In other words, we set $\theta = X$ and $\gamma = \phi$. In Paper IV, we reverse the roles of X and ϕ . That is, we use EM to estimate the data association ϕ , from which we can obtain the state estimates X immediately.

Chapter 5

Metrics

Metrics are important in multi-target tracking (MTT) for performance evaluation and algorithm design. In essence, metrics are necessary to quantify the closeness between a ground truth and an estimate thereof. When designing metrics for MTT, there are specific challenges that must be addressed, such as localisation error, error due to missed and false targets and penalty for track switches.

In this chapter, we discuss the need for metrics in MTT, followed by a discussion on the basic metric properties. We also present a summary on the commonly used metrics, and briefly discuss the challenges in designing a metric for trajectory estimation.

5.1 Need for a metric

Metrics are needed in trajectory estimation for two main reasons: designing algorithms and performance evaluation. In algorithm design, one wants an estimate that is close to the true state in some sense. It is reasonable that a metric is used to define this closeness. When evaluating the performance of an algorithm, one needs a similarity measure to quantify the error between the obtained estimates and the ground truth. Once again, it seems natural that a metric is used to quantify this error.

In multi-target tracking (MTT), the estimation is often formulated as a Bayesian filtering problem where the ground truth is a random quantity and the estimates depend deterministically on the observed data. For performance evaluation, in many cases, we average over several realizations of the data, so estimates are random as well. In both the scenarios we discussed above—designing algorithms and performance evaluation—the objectives

involve an averaging of the metric over different instances of the random quantities involved, i.e., the ground truth and the estimate. In other words, the similarity between the ground truth and the estimate is computed in an average sense. It is again important that this similarity which involves averaging is also a metric.

5.2 Metric properties

The definition of a metric varies slightly based on if the variables involved are random or not. In this section, we summarize the properties of a metric on general spaces and on probability spaces. We also discuss the significance of these properties briefly.

Definition 5.1. *A metric $d_A(\cdot, \cdot)$ on a set A is a function that satisfies the following properties for any $x, y, z \in A$ [73, Sec. 2.15]:*

1. *Non-negativity:* $d_A(x, y) \geq 0$.
2. *Definiteness:* $d_A(x, y) = 0 \Leftrightarrow x = y$.
3. *Symmetry:* $d_A(x, y) = d_A(y, x)$.
4. *Triangle inequality:* $d_A(x, y) \leq d_A(x, z) + d_A(z, y)$.

For metrics in a probability space \mathcal{A} , the definiteness between random variables is in the almost-sure sense [74, Sec. 2.2], as described in the following definition.

Definition 5.2. *A metric $d_{\mathcal{A}}(\cdot, \cdot)$ on a set \mathcal{A} is a function that satisfies the following properties for random variables $x, y, z \in \mathcal{A}$:*

1. *Non-negativity:* $d_{\mathcal{A}}(x, y) \geq 0$.
2. *Definiteness:* $d_{\mathcal{A}}(x, y) = 0 \Leftrightarrow \Pr(x = y) = 1$.
3. *Symmetry:* $d_{\mathcal{A}}(x, y) = d_{\mathcal{A}}(y, x)$.
4. *Triangle inequality:* $d_{\mathcal{A}}(x, y) \leq d_{\mathcal{A}}(x, z) + d_{\mathcal{A}}(z, y)$.

In the above definition, $\Pr(x = y) = 1$ implies that the event that x and y take the same value has probability 1.

We now proceed to describe and discuss the significance of these properties, with more emphasis on the triangle inequality property. The non-negativity property ensures that the distance cannot be negative. The definiteness property ensures that a distance between a point to itself is 0. For

5.3. COMMON METRICS

random variables, this property is in essence ensured for all the points that have non-zero probability. The symmetry property confirms that the distance from point x to y should be the same as the distance from point y to x .

The triangle inequality property, despite its abstractness, has a major practical importance in algorithm assessment [75, Sec. 6.2.1]. Suppose there are two estimates y and z for a ground truth x . Let us assume that according to $d_{\mathcal{A}}$, the estimate z is close to the ground truth x and is also close to the other estimate y . Then, according to intuition, the second estimate y should also be close to the ground truth x . This property is ensured by the triangle inequality property. The triangle inequality also has practical implications to ensure the quality of approximate optimal estimators. Consider x to be the ground truth, and z to be the optimal estimate, according to a certain criterion. Let us assume that it is difficult to compute the optimal estimate z such that we resort to an approximation y of the optimal z . This happens often in practice. If the triangle inequality does not hold, it would mean that even if we have a good estimate y , close to the optimal z , which in turn is close to the ground truth x , it is possible that the distance from y to the ground truth x is high. This property is clearly not desirable.

5.3 Common metrics

In this section, we discuss some of the commonly used metrics in the literature. One way of categorizing the metrics is based on the kind of variables involved. Below, we summarize the metrics used for vectors, finite sets of vectors and finite sets of trajectories. Before we present the metrics, we first discuss in each section in which scenarios these kinds of metrics are useful.

5.3.1 On vector spaces

Metric on vector spaces is a well studied problem with the most common one being the Euclidean metric. The root mean square error (RMSE) is based on the Euclidean metric and is commonly used when the involved quantities are random. Below we present a slightly general version of the RMSE.

Definition 5.3. *Given two vectors x, y in \mathbb{R}^N , then the p -norm for any $1 \leq p \leq \infty$ is a metric. It is defined as follows:*

$$d_p(x, y) \triangleq \sqrt[p]{\sum_{i=1}^N |x_i - y_i|^p}. \quad (5.1)$$

Definition 5.4. *If the vectors x and y are random vectors with joint distribution $f(x, y)$, then the following definition is also a metric:*

$$\bar{d}_p(x, y) \triangleq \sqrt[p]{\mathbb{E}[d_p(x, y)^p]}, \quad (5.2)$$

where the expectation is defined with respect to the joint distribution $f(x, y)$. When we set $p = 2$ in the above definition, we get the commonly used RMSE metric.

The Euclidean metric is also commonly used in the trajectory estimation problem for simple scenarios. For instance, in the single trajectory estimation problem, where there is no uncertainty in the birth time and the death time of the trajectory, then there is a one-to-one correspondence between the estimated state and the ground truth at each time instant. In this case, one can just use the metric for vectors at each time instant.

5.3.2 On the space of finite sets of vectors

Let us now consider scenarios with multiple targets, where we are interested in how good the localisation is at each time instant. In this case, at each time instant, both the ground truth and the estimates are sets of state vectors, where there is uncertainty about which vector in the estimated set corresponds to which vector in the ground truth. Now, the quantity of interest is a metric between sets of vectors.

The study of the metrics in this space is relatively new. In the MTT literature, there are several metrics that have been proposed for this purpose, such as the Wasserstein metric [75, 76], the Hausdorff metric [76], the OSPA metric [77] and many more [78–83]. Among these, the optimal sub-pattern assignment (OSPA) metric is the most commonly used one. The metrics in [82] and [83] propose a base distance in the metric that also takes into the account the quality information about the estimated state. Below we present the OSPA metric.

Definition 5.5. *Let $d(\cdot, \cdot)$ denote a metric on \mathbb{R}^N such that $d(x, y)$ is the distance between $x, y \in \mathbb{R}^N$ and let $d^{(c)}(x, y) = \min(d(x, y), c)$ be the cut-off metric associated with $d(x, y)$ [75, Sec. 6.2.1]. We also refer to $d^{(c)}(x, y)$ as base distance. Let Π_n be the set of all permutations in $\{1, \dots, n\}$ for any $n \in \mathbb{N}$. Any element $\pi \in \Pi_n$ is a sequence $(\pi(1), \dots, \pi(n))$.*

Let $X = \{x_1, \dots, x_{|X|}\}$ and $Y = \{y_1, \dots, y_{|Y|}\}$ be finite subsets of a bounded observation window $W \subset \mathbb{R}^N$, where $|A|$ denotes the cardinality of

5.3. COMMON METRICS

a set A . For $1 \leq p < \infty$ and $|X| \leq |Y|$, OSPA [77] is defined as

$$d_p^{(c)}(X, Y) \triangleq \left(\frac{1}{|Y|} \left(\min_{\pi \in \Pi_{|Y|}} \sum_{i=1}^{|X|} d^{(c)}(x_i, y_{\pi(i)})^p + c^p(|Y| - |X|) \right) \right)^{\frac{1}{p}}. \quad (5.3)$$

For $|X| > |Y|$, $d_p^{(c)}(X, Y) = d_p^{(c)}(Y, X)$. The ∞ -OSPA is defined as

$$d_\infty^{(c)}(X, Y) \triangleq \begin{cases} \min_{\pi \in \Pi_{|Y|}} \max_{1 \leq i \leq |X|} d^{(c)}(x_i, y_{\pi(i)}) & |X| = |Y| \\ c & \text{otherwise} \end{cases}. \quad (5.4)$$

In Paper V [84], we discuss the shortcomings of the above formulation. We propose a new metric which addresses localisation error as well as missed and false targets that are of interest in MTT. We also extend the metric to compute the equivalents of the RMSE metric for vectors.

5.3.3 On the space of finite sets of trajectories

In many tracking algorithms, such as in multiple hypothesis tracking (MHT) [27, 46, 47] and joint probabilistic data association (JPDA) [49, 50], the output of the algorithm is not just the set of states at each time. Instead, the output is a set of time sequences of states, i.e., trajectories of states. Note that the theory for sets of trajectories has been well established in [85]. To define a metric between sets of trajectories, it is common to use the metric discussed in Section 5.3.2 or a simpler modification of it. But this strategy produces strange and counter-intuitive behavior. Below, we discuss some of those approaches and their shortcomings.

One approach is to use OSPA on the entire sets of trajectories [86, 87]. In this approach, one uses the OSPA definition where the base metric between two tracks is defined. To discuss the problems with this approach, let us consider a new set of examples in Figures 5.1 and 5.2. The ground truth is the trajectory shown in blue o's in both the figures and the estimates are the ones shown in red x's. According to the metric we just discussed, OSPA picks the one with the minimum of the base distance between the tracks in the ground truth and the tracks in the estimates. Assuming ϵ is almost 0 and ρ is large, the OSPA distance indicates that both the estimates in Figure 5.1 and 5.2 have the same distance to the ground truth. This is clearly counter-intuitive. The estimate in Figure 5.1 is clearly a better one compared to the one in Figure 5.2. This undesirable behavior is due to the property that OSPA assigns each track in the ground truth to exactly one track in the estimate, assuming the estimate has more tracks than the

ground truth.

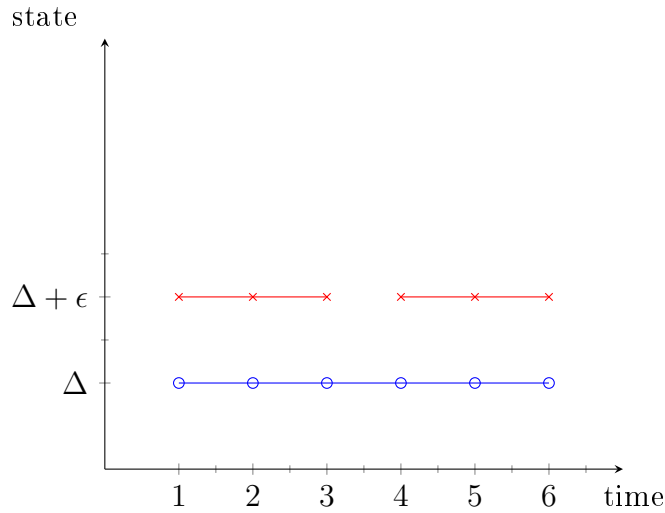


Figure 5.1: If ϵ is small, the estimate indicated by red \times 's is still a good estimate compared to the one in Figure 5.2 for the ground truth in blue o's. The only problem with this estimate is that it has split a single trajectory into two.

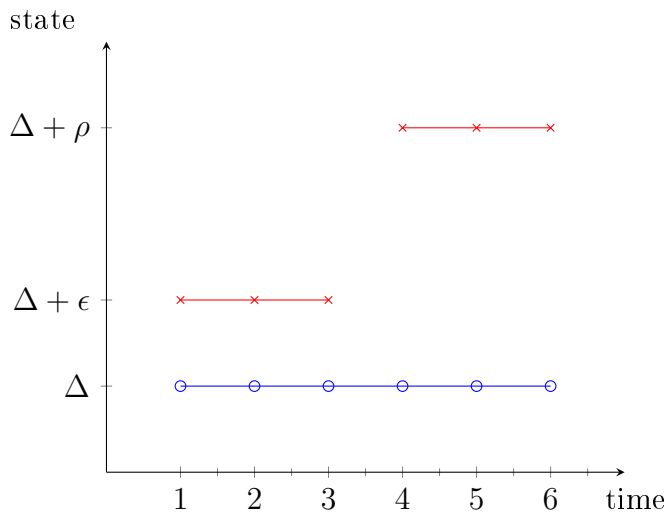


Figure 5.2: If ϵ is small and ρ is large, the estimate tracks in red color with \times 's should have a larger distance to the ground truth in blue o's compared to the estimate in Figure 5.1.

It is common to directly use the OSPA metric on the set of states of the trajectories at each time instant. A shortcoming of this approach can be illustrated with a simple example. Let us assume that in the ground truth we have a trajectory of states, i.e., a time sequence of states. Let us consider two different version of the estimates:

5.3. COMMON METRICS

- Estimate 1: we obtain a time sequence of states exactly identical to the one in the ground truth,
- Estimate 2: we obtain two time sequences of states, one which exactly matches the first half of the track in the ground truth and the second sequence which exactly matches the second half of the track in the ground truth. (This corresponds to $\epsilon = 0$ in the example in Figure 5.1).

Using the strategy we just discussed, we get the exact same value, 0, for the ‘metric’ for both these estimates. This property clearly violates the definiteness property we discussed in the beginning of the chapter.

A summary of the learnings from the above two approaches is that you can assign a track in the ground truth to different tracks at different times, but there should be an additional cost for being assigned to different tracks. This we refer to as the switching cost. We have used this approach in Paper VI of the thesis. In the paper [88], we compare our metric to some of the other metrics [89,90] in the literature that uses the same approach.

Chapter 6

Contributions and future work

The main objectives of the thesis are to design algorithms for addressing the data association problem in trajectory estimation and to design a metric to evaluate the algorithms. The contributions of each paper comprising the thesis are discussed briefly in this chapter. Furthermore, possible ideas for future research that arose during the writing of this thesis are presented.

6.1 Contributions

In the following section, the contributions of the six papers in the thesis, and the relations between them, are presented.

Paper I

In this paper, the problem of forward-backward smoothing (FBS) of Gaussian mixture (GM) densities based on merging is addressed. The existing literature provides practical algorithms for the FBS of GMs that are based on pruning. The drawback of a pruning strategy is that as a result of excessive pruning, the forward filtering can result in degeneracy. The backward smoothing on this degenerate forward filter can lead to highly underestimated data association uncertainties. To overcome this, we propose using merging of the GM during forward filtering as well as during backward smoothing. As mentioned before, the forward filter based on merging is well studied in the literature. A strategy to perform the backward smoothing on filtered densities with merged components is analysed and explained in this paper. When compared to FBS based on an N-scan pruning algorithm, the two-filter smoothed densities obtained using the presented approximations of the backward filter show better track-loss, root mean squared error (RMSE) and normalized estimation error squared (NEES) for lower complexity.

Paper II

The objective of this paper is to obtain an algorithm for two-filter smoothing (TFS) of GM densities based on merging approximations. The TFS involves two filters, namely the forward filter and the backward filter, where the former has been studied extensively in the literature. The latter, i.e., the backward filter, has a structure similar to a GM, but is not a normalizable density. Therefore, the traditional Gaussian mixture reduction (GMR) algorithms cannot be applied directly in the backward filter. The existing literature, though providing an analysis of the backward filter, does not present a strategy for the involved GMR. This paper presents two strategies using which the Gaussian mixture reduction (GMR) can be applied to the backward filter. The first one is an intragroup approximation method which depends on the structure of the backward filter, and presents a way in which GMR can be applied within certain groups of components. The second method is a smoothed posterior pruning method, which is similar to the pruning strategy for the (forward) filtered densities discussed in [91]. In Paper I, the posterior pruning idea is formulated and proved to be a valid operation for both the forward and the backward filters. When compared to FBS based on an N-scan pruning algorithm, the two-filter smoothed densities obtained using the presented approximations of the backward filter are shown to have better track-loss, RMSE and NEES for lower complexity.

Paper III

This paper address the data association problem in multiple trajectory estimation. The objective in this paper is to obtain the maximum a posteriori (MAP) estimate of the state X from the joint density $p(X, \phi, Z)$. This problem is not tractable as the number of possibilities of ϕ is exponential. In this paper, we address the problem using expectation maximisation (EM) to estimate X , while treating ϕ as a hidden variable. We show that state estimates can be obtained by running an iterative algorithm, where in each iteration, a Rauch-Tung Striebel (RTS) smoother is run for each target. The measurements updates in the filter and smoother is carried out with the composite measurements, which are weighted sums of the measurements at each time instant. The weights, which are the marginal data association probabilities, are computed using loopy belief propagation. We show in the paper, that despite the simplicity, the algorithm performance is comparable to a multiple hypothesis tracking (MHT) algorithm.

Paper IV

The data association problem is addressed in this paper by estimating the data association variable ϕ from the joint density $p(X, \phi, Z)$. Once ϕ is estimated, X is immediate to estimate using an RTS smoother. The strategy is to use EM to estimate ϕ . This strategy results in an iterative algorithm, where in each iteration, one runs an RTS smoother for each target. The measurements for the RTS smoother are obtained using global nearest neighbour (GNN) at each time. In the paper, we show that the algorithm outperforms an MHT implementation in terms of mean optimal sub-pattern assignment (OSPA) performance.

Paper V

In this paper, we present a metric named generalised OSPA (GOSPA) to compute distance between two sets of vectors. We show that compared to the OSPA metric, our metric addresses the problem by penalising missed and false targets, whereas OSPA penalises the cardinality mismatch. We also show that the GOSPA metric can be extended to random finite sets of vectors, which is relevant for performance evaluation and algorithm design. We show that given a joint distribution over two sets of vectors, the mean GOSPA and the root mean squared GOSPA are also metrics.

Paper VI

In this paper, we propose a metric based on multidimensional assignments in the space of sets of trajectories. Besides the localisation cost, missed and false targets [84], this metric also addresses the problems of track switches by allowing a trajectory to be assigned to multiple trajectories across time, but by penalising it for these multiple assignments. We introduce the concepts of half and full switches to quantify the penalty. As this multidimensional assignment metric belongs to the NP hard class of problems, we also propose a lower bound for the metric, which is computable in polynomial time using linear programming (LP). We also show that this lower bound is a metric in the space of sets of trajectories. From simulations, we have observed that the lower bound computed using LP often returns the optimal value for the multidimensional metric. An efficient way to compute the LP metric using alternating direction method of multipliers (ADMM) that scales linearly with time is also presented. We further adapt this metric to random sets of trajectories.

6.2 Future work

Besides the ideas and algorithms presented in the thesis, we also obtained a plethora of ideas to investigate in the future. In this section, we present and discuss the ideas, which range from the extensions of GM smoothing to more complex scenarios than single-target linear Gaussian process models, to computationally cheaper GM merging methods and message passing in generic graphs.

Merging algorithms

The TFS and FBS algorithms presented in the thesis are based on merging. There are several methods, such as Runnalls', Salmond's and variants of these, which one can choose for GM merging implementation. However, the computational complexity of these methods is a serious limitation when it comes to practical implementations where GM merging is necessary at each time instant. In both reduction algorithms, the merging cost must be computed for every pair of components, which involves expensive matrix multiplications. Therefore, the complexity of these algorithms is quadratic, if not exponential, in the number of components, which is still expensive considering prediction, update and retrodiction steps. For the results presented in the thesis, significant amount of effort went into devising practical merging algorithms, which resulted in two strategies. One is a combination of Runnalls' and Salmond's algorithms, which is used in the forward filter. The other method is a modified version of Salmond's algorithm. A possible investigation can be in making fewer cost computations than computing the cost for every pair (i, j) . One way of reducing the number of merging cost computations is by obtaining bounds on the cost function. Suppose there is an upper bound on the least possible cost. And suppose that for some group of pairs of components, we can compute a lower bound on the costs. If the group's lower bound is greater than the upper bound on the lowest cost, the cost computation for the component pairs in the group can be avoided. The challenge is thus in obtaining the upper bound on the least cost, and selecting the group that can be eliminated. A closer analysis of the cost function is necessary to obtain these bounds and a good choice of groups.

Trajectory estimation with random birth and death events

In Paper III and Paper IV, it is assumed that all the tracks are present the whole batch duration. It would be interesting to extend the approach to cases where the tracks births and deaths happen at random times. One

6.2. FUTURE WORK

exhaustive approach is to consider all possible birth and death time combinations for all the tracks. Similar to the data association problem, this is also a combinatorial problem. A more appealing approach would be to model these variables into the joint density and use EM to estimate the birth and death time variables as well.

Online algorithms

The algorithms proposed in Papers I to IV are all batch algorithms. Extending all these algorithms to online algorithms extends the scope of these algorithms. There are several possibilities to investigate here. For instance, one can use a sliding window approach, where one can tune the width of the window and also the overlap across the windows based on the application. Another approach is to extend the idea of smoothed filtering proposed in [91]. That is, to obtain the filtered density $p(x_k|z_{1:k})$, one can go back and improve all the approximations made at all the previous time instants. This improvement in approximation can be implemented using an iterative approach in the papers.

Metric for sets of trajectories based on distance-based switching cost

In Paper VI, track switches are used to penalise when a trajectory in the ground truth set is assigned to multiple trajectories in the estimate. In the current version, the penalty for the track switch is a fixed parameter. But there can be scenarios when this penalty must be varied based on the severity of the switch. For instance, consider a pair of trajectories in the ground truth that are close to each other for certain duration and then move apart. Let us consider two estimates for the ground truth. First is an estimate with trajectories such that the track switch happens when the trajectories in the ground truth are close together. The second estimate has trajectories such that the track switch happens when the two trajectories in the ground truth are far apart. According to intuition, the first estimate is better than the second, as the track switch happens in a region where it might be difficult to resolve. This difference should be possible to address by defining a penalty for the track switch that depends on the closeness of trajectories in the ground truth and their corresponding trajectories in the estimate. The major challenge here can be in defining the penalty in a consistent way that still retains the metric properties. This would be an interesting problem to investigate.

References

- [1] B. D. Anderson and J. B. Moore, *Optimal filtering*. Courier Dover Publications, 2012.
- [2] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Journal of basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.
- [3] W. Koch and F. Govaers, “On accumulated state densities with applications to out-of-sequence measurement processing,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 4, pp. 2766–2778, 2011.
- [4] H. E. Rauch, C. Striebel, and F. Tung, “Maximum likelihood estimates of linear dynamic systems,” *AIAA journal*, vol. 3, no. 8, pp. 1445–1450, 1965.
- [5] M. Briers, A. Doucet, and S. Maskell, “Smoothing algorithms for state–space models,” *Annals of the Institute of Statistical Mathematics*, vol. 62, no. 1, pp. 61–89, 2010.
- [6] K. Ito and K. Xiong, “Gaussian filters for nonlinear filtering problems,” *IEEE Transactions on Automatic Control*, vol. 45, no. 5, pp. 910–927, 2000.
- [7] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, “A new approach for filtering nonlinear systems,” in *Proceedings of the American Control Conference*, vol. 3. IEEE, 1995, pp. 1628–1632.
- [8] S. J. Julier and J. K. Uhlmann, “Unscented filtering and nonlinear estimation,” *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.
- [9] I. Arasaratnam, S. Haykin, and R. J. Elliott, “Discrete-time nonlinear filtering algorithms using Gauss–Hermite quadrature,” *Proceedings of the IEEE*, vol. 95, no. 5, pp. 953–977, 2007.
- [10] I. Arasaratnam and S. Haykin, “Cubature Kalman filters,” *IEEE Transactions on Automatic Control*, vol. 54, no. 6, pp. 1254–1269, 2009.

REFERENCES

- [11] A. Doucet and A. M. Johansen, “A tutorial on particle filtering and smoothing: Fifteen years later,” *Handbook of Nonlinear Filtering*, vol. 12, pp. 656–704, 2009.
- [12] W. K. Hastings, “Monte Carlo sampling methods using Markov chains and their applications,” *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.
- [13] A. Smith, A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo methods in practice*. Springer Science & Business Media, 2013.
- [14] A. Doucet, S. Godsill, and C. Andrieu, “On sequential Monte Carlo sampling methods for Bayesian filtering,” *Statistics and computing*, vol. 10, no. 3, pp. 197–208, 2000.
- [15] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking,” *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.
- [16] N. Gordon, B. Ristic, and S. Arulampalam, *Beyond the Kalman filter: Particle filters for tracking applications*. Artech House, London, 2004.
- [17] D. Fraser and J. Potter, “The optimum linear smoother as a combination of two optimum linear filters,” *IEEE Transactions on Automatic Control*, vol. 14, no. 4, pp. 387–390, 1969.
- [18] A. H. Jazwinski, *Stochastic processes and filtering theory*. Courier Dover Publications, 2007.
- [19] Á. F. García-Fernández, L. Svensson, M. Morelande, and S. Särkkä, “Posterior linearization filter: Principles and implementation using sigma points,” *IEEE Transactions on Signal Processing*, vol. 63, no. 20, 2015.
- [20] S. S. Haykin, S. S. Haykin, and S. S. Haykin, *Kalman filtering and neural networks*. Wiley Online Library, 2001.
- [21] E. A. Wan, R. Van Der Merwe, and A. T. Nelson, “Dual estimation and the unscented transformation.” in *NIPS*, 1999, pp. 666–672.
- [22] S. Sarkka, “Unscented Rauch–Tung–Striebel smoother,” *IEEE Transactions on Automatic Control*, vol. 53, no. 3, pp. 845–849, 2008.
- [23] C. Andrieu, A. Doucet, and R. Holenstein, “Particle Markov chain Monte Carlo methods,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 72, no. 3, pp. 269–342, 2010.

REFERENCES

- [24] G. Kitagawa, “Monte Carlo filter and smoother for non Gaussian non-linear state space models,” *Journal of computational and graphical statistics*, vol. 5, pp. 1–25, 1996.
- [25] P. Fearnhead, D. Wyncoll, and J. Tawn, “A sequential smoothing algorithm with linear computational cost,” *Biometrika*, vol. 97, no. 2, pp. 447–464, 2010.
- [26] S. J. Godsill, A. Doucet, and M. West, “Monte Carlo smoothing for nonlinear time series,” *Journal of the American Statistical Association*, vol. 99, no. 465, 2004.
- [27] S. S. Blackman and R. Popoli, *Design and analysis of modern tracking systems*. Artech House, 1999, vol. 685.
- [28] K. G. Murty, “Letter to the editor: An algorithm for ranking all the assignments in order of increasing cost,” *Operations Research*, vol. 16, no. 3, pp. 682–687, 1968.
- [29] I. J. Cox and S. L. Hingorani, “An efficient implementation of Reid’s multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 2, pp. 138–150, 1996.
- [30] I. J. Cox and M. L. Miller, “On finding ranked assignments with application to multitarget tracking and motion correspondence,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 31, no. 1, pp. 486–489, 1995.
- [31] D. Salmond, “Mixture reduction algorithms for point and extended object tracking in clutter,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 2, pp. 667–686, 2009.
- [32] D. J. Salmond, “Mixture reduction algorithms for uncertain tracking,” DTIC Document, Tech. Rep., 1988.
- [33] ———, “Mixture reduction algorithms for target tracking,” in *IEE Colloquium on State Estimation in Aerospace and Tracking Applications*. IET, 1989, pp. 7–1.
- [34] A. R. Runnalls, “Kullback-Leibler approach to Gaussian mixture reduction,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 3, pp. 989–999, 2007.

REFERENCES

- [35] J. L. Williams and P. S. Maybeck, “Cost-function-based Gaussian mixture reduction for target tracking,” in *Proceedings of the 6th International Conference on Information Fusion*, vol. 2, 2003, pp. 1047–1054.
- [36] D. Crouse, P. Willett, K. Pattipati, and L. Svensson, “A look at Gaussian mixture reduction algorithms,” in *Proceedings of the 14th International Conference on Information Fusion*, 2011, pp. 1–8.
- [37] D. J. Salmond, “Mixture reduction algorithms for target tracking in clutter,” in *Proceedings of SPIE: Signal and Data Processing of Small Targets Conference*. International Society for Optics and Photonics, 1990, pp. 434–445.
- [38] D. J. Petrucci, “Gaussian mixture reduction for Bayesian target tracking in clutter,” Master’s thesis, Air Force Institute of Technology, 2005. [Online]. Available: <http://www.dtic.mil/dtic/tr/fulltext/u2/a443588.pdf>
- [39] J. L. Williams, “Gaussian mixture reduction for tracking multiple maneuvering targets in clutter,” Master’s thesis, Air Force Institute of Technology, 2003.
- [40] S. Kullback and R. A. Leibler, “On information and sufficiency,” *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.
- [41] M. F. Huber and U. D. Hanebeck, “Progressive Gaussian mixture reduction,” in *Proceedings of the 11th International Conference on Information Fusion*. IEEE, 2008, pp. 1–8.
- [42] D. W. Scott, “Parametric statistical modeling by minimum integrated square error,” *Technometrics*, vol. 43, no. 3, pp. 274–285, 2001.
- [43] D. W. Scott and W. F. Szewczyk, “From kernels to mixtures,” *Technometrics*, vol. 43, no. 3, pp. 323–335, 2001.
- [44] A. S. Rahmathullah, L. Svensson, and D. Svensson, “Merging-based forward-backward smoothing on Gaussian mixtures,” in *Proceedings of the 17th International Conference on Information Fusion*. IEEE, 2014, pp. 1–8.
- [45] ———, “Two-filter Gaussian mixture smoothing with posterior pruning,” in *Proceedings of the 17th International Conference on Information Fusion*. IEEE, 2014, pp. 1–8.
- [46] D. Reid, “An algorithm for tracking multiple targets,” *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 843–854, 1979.

REFERENCES

- [47] T. Kurien and M. Liggins, “Report-to-target assignment in multisensor multitarget tracking,” in *Proceedings of the 27th IEEE Conference on Decision and Control*. IEEE, 1988, pp. 2484–2488.
- [48] O. E. Drummond, “Multiple target tracking with multiple frame, probabilistic data association,” in *Optical Engineering and Photonics in Aerospace Sensing*. International Society for Optics and Photonics, 1993, pp. 394–408.
- [49] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe, “Multi-target tracking using joint probabilistic data association,” in *IEEE Conference on Decision and Control including the Symposium on Adaptive Processes*, vol. 19. IEEE, 1980, pp. 807–812.
- [50] K.-C. Chang and Y. Bar-Shalom, “Joint probabilistic data association for multitarget tracking with possibly unresolved measurements and maneuvers,” *IEEE Transactions on Automatic Control*, vol. 29, no. 7, pp. 585–594, 1984.
- [51] L. Svensson, D. Svensson, M. Guerriero, and P. Willett, “Set JPDA filter for multitarget tracking,” *IEEE Transactions on Signal Processing*, vol. 59, no. 10, pp. 4677–4691, 2011.
- [52] L. Y. Pao, “Multisensor multitarget mixture reduction algorithms for tracking,” *Journal of Guidance, Control, and Dynamics*, vol. 17, no. 6, pp. 1205–1211, 1994.
- [53] Y. Bar-Shalom, F. Daum, and J. Huang, “The probabilistic data association filter,” *Control Systems Magazine*, vol. 29, no. 6, pp. 82–100, 2009.
- [54] J. L. Williams and R. A. Lau, “Data association by loopy belief propagation,” in *Proceedings of the 13th International Conference on Information Fusion*, 2010.
- [55] J. Williams and R. Lau, “Approximate evaluation of marginal association probabilities with belief propagation,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 4, pp. 2942–2959, 2014.
- [56] R. L. Streit and T. E. Luginbuhl, “Probabilistic multi-hypothesis tracking,” DTIC Document, Tech. Rep., 1995.
- [57] P. Willett, Y. Ruan, and R. Streit, “PMHT: Problems and some solutions,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 3, pp. 738–754, 2002.

REFERENCES

- [58] D. F. Crouse, M. Guerriero, and P. Willett, "A critical look at theP-MHT," *Journal of Advances in Information Fusion*, vol. 4, no. 2, pp. 93–116, 2009.
- [59] K. J. Molnar and J. W. Modestino, "Application of the EM algorithm for the multitarget/multisensor tracking problem," *IEEE Transactions on Signal Processing*, vol. 46, no. 1, pp. 115–129, 1998.
- [60] S. Oh, S. Russell, and S. Sastry, "Markov chain Monte Carlo data association for general multiple-target tracking problems," in *Proceedings of the IEEE Conference on Decision and Control*, 2004.
- [61] H. W. Kuhn, "Variants of the Hungarian method for assignment problems," *Naval Research Logistics Quarterly*, vol. 3, no. 4, pp. 253–258, 1956.
- [62] D. P. Bertsekas, "The auction algorithm: A distributed relaxation method for the assignment problem," *Annals of operations research*, vol. 14, no. 1, pp. 105–123, 1988.
- [63] R. Jonker and A. Volgenant, "A shortest augmenting path algorithm for dense and sparse linear assignment problems," *Computing*, vol. 38, no. 4, pp. 325–340, 1987.
- [64] S. Deb, M. Yeddanapudi, K. Pattipati, and Y. Bar-Shalom, "A generalized SD assignment algorithm for multisensor-multitarget state estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 2, pp. 523–538, 1997.
- [65] A. B. Poore, A. J. Robertson III, and P. J. Shea, "New class of Lagrangian-relaxation-based algorithms for fast data association in multiple hypothesis tracking applications," in *SPIE Symposium on OE/Aerospace Sensing and Dual Use Photonics*. International Society for Optics and Photonics, 1995, pp. 184–194.
- [66] R. L. Popp, K. R. Pattipati, and Y. Bar-Shalom, "m-best SD assignment algorithm with application to multitarget tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 1, pp. 22–39, 2001.
- [67] I. J. Cox and M. L. Miller, "A comparison of two algorithms for determining ranked assignments with application to multitarget tracking and motion correspondence," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 1, pp. 295–301, 1997.

REFERENCES

- [68] T. Minka, “Divergence measures and message passing,” Technical report, Microsoft Research, Tech. Rep., 2005.
- [69] A. S. Rahmathullah, A. F. García-Fernández, and L. Svensson, “A batch algorithm for estimating trajectories of point targets using expectation maximization,” *To be published in IEEE Transactions on Signal processing*, 2016.
- [70] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.
- [71] M. J. Beal, “Variational algorithms for approximate Bayesian inference,” Ph.D. dissertation, University of London, 2003.
- [72] A. S. Rahmathullah, A. F. García-Fernández, and L. Svensson, “Expectation maximization with hidden state variables for multi-target tracking,” *To be submitted for publication to IEEE Transactions on Signal processing*, 2016.
- [73] W. Rudin, *Principles of mathematical analysis*. McGraw-Hill New York, 1964, vol. 3.
- [74] S. T. Rachev, L. Klebanov, S. V. Stoyanov, and F. Fabozzi, *The methods of distances in the theory of probability and statistics*. Springer Science & Business Media, 2013.
- [75] R. P. Mahler, *Advances in statistical multisource-multitarget information fusion*. Artech House, 2014.
- [76] J. R. Hoffman and R. P. Mahler, “Multitarget miss distance via optimal assignment,” *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, vol. 34, no. 3, pp. 327–336, 2004.
- [77] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, “A consistent metric for performance evaluation of multi-object filters,” *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3447–3457, 2008.
- [78] B. E. Fridling and O. E. Drummond, “Performance evaluation methods for multiple-target-tracking algorithms,” in *Orlando’91, Orlando, FL*. International Society for Optics and Photonics, 1991, pp. 371–383.
- [79] O. E. Drummond and B. E. Fridling, “Ambiguities in evaluating performance of multiple target tracking algorithms,” in *Aerospace Sensing*. International Society for Optics and Photonics, 1992, pp. 326–337.

REFERENCES

- [80] O. Fujita, “Metrics based on average distance between sets,” *Japan Journal of Industrial and Applied Mathematics*, vol. 30, no. 1, pp. 1–19, 2013.
- [81] A. Gardner, J. Kanno, C. Duncan, and R. Selmic, “Measuring distance between unordered sets of different sizes,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 137–143.
- [82] X. He, R. Tharmarasa, T. Kirubarajan, and T. Thayaparan, “A track quality based metric for evaluating performance of multitarget filters,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 1, pp. 610–616, 2013.
- [83] S. Nagappa, D. E. Clark, and R. Mahler, “Incorporating track uncertainty into the OSPA metric,” in *Proceedings of the 14th International Conference on Information Fusion*. IEEE, 2011, pp. 1–8.
- [84] A. S. Rahmathullah, A. F. García-Fernández, and L. Svensson, “Generalized optimal sub-pattern assignment metric,” *Submitted for publication to IEEE Transactions on Signal processing*, 2016. [Online]. Available: <http://arxiv.org/pdf/1601.05585v1.pdf>
- [85] L. Svensson and M. Morelande, “Target tracking based on estimation of sets of trajectories,” in *Proceedings of the 17th International Conference on Information Fusion*. IEEE, 2014.
- [86] B. Ristic, B.-N. Vo, and D. Clark, “Performance evaluation of multi-target tracking using the OSPA metric,” in *Proceedings of the 13th International Conference on Information Fusion*, 2010.
- [87] B. Ristic, B.-N. Vo, D. Clark, and B.-T. Vo, “A metric for performance evaluation of multi-target tracking algorithms,” *IEEE Transactions on Signal Processing*, vol. 59, no. 7, pp. 3452–3457, 2011.
- [88] A. S. Rahmathullah, A. F. García-Fernández, and L. Svensson, “A metric on the space of finite sets of trajectories for evaluation of multi-target tracking algorithms,” *Submitted for publication to IEEE Transactions on Signal processing*, 2016.
- [89] T. Vu and R. Evans, “A new performance metric for multiple target tracking based on optimal subpattern assignment,” in *Proceedings of the 17th International Conference on Information Fusion*. IEEE, 2014.

REFERENCES

- [90] J. Bento, “A metric for sets of trajectories that is practical and mathematically consistent,” *arXiv preprint arXiv:1601.03094*, 2016.
- [91] A. S. Rahmathullah, L. Svensson, D. Svensson, and P. Willett, “Smoothed probabilistic data association filter,” in *Proceedings of the 16th International Conference on Information Fusion*. IEEE, 2013, pp. 1296–1303.