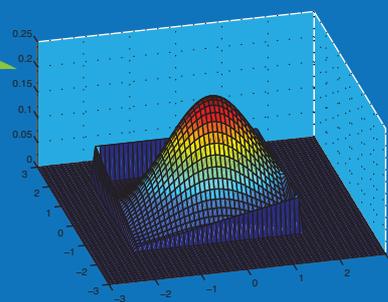
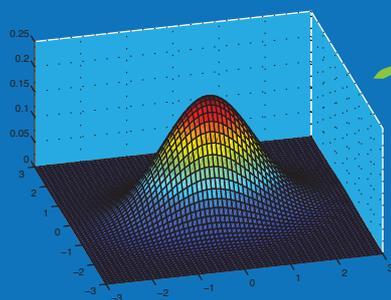
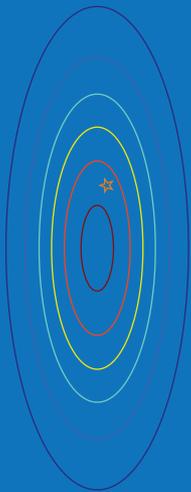
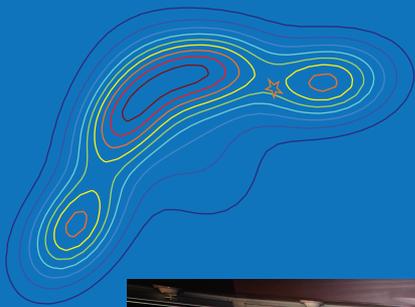


# Nonlinear State and Parameter Estimation for Hopper Dredgers

*Pawel Mirosław Stano*





# Nonlinear State and Parameter Estimation for Hopper Dredgers

Paweł Mirosław Stano

The cover illustration: the control room of the modern Trailing Suction Hopper Dredger with the view of the Rotterdam harbour. Courtesy of IHC Systems.

# Nonlinear State and Parameter Estimation for Hopper Dredgers

## Proefschrift

ter verkrijging van de graad van doctor  
aan de Technische Universiteit Delft,  
op gezag van de Rector Magnificus prof. ir. K.C.A.M. Luyben  
voorzitter van het College voor Promoties,  
in het openbaar te verdedigen op  
woensdag 12 juni 2013 om 12:30 uur  
door

**Paweł Mirosław Stano**

Master of Science,  
geboren te Kraków, Polen.

Dit proefschrift is goedgekeurd door de promotor:  
Prof. dr. R. Babuška

Samenstelling promotiecommissie:

Rector Magnificus,	voorzitter
Prof. dr. R. Babuška	Technische Universiteit Delft, promotor
Prof. dr. ir. C. van Rhee	Technische Universiteit Delft
Prof. dr. ir. J. H. van Schuppen	Technische Universiteit Delft
Prof. dr. ir. G. Leus	Technische Universiteit Delft
Dr. hab. W. Słomczyński	Jagiellonian University, Kraków
Dr. Zs. Lendek	Technical University of Cluj-Napoca
Dr. ir. J. Braaksma	Huisman Equipment B.V.
Prof. dr. ir. B. De Schutter	Technische Universiteit Delft, reservelid

Copyright © 2013 by Paweł M. Stano.

All rights reserved. No part of the material protected by this copyright notice may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage and retrieval system, without written permission from the copyright owner.

Printed in the Netherlands

*“If a person is endowed with sensibility and has a deep enough insight into the workings of the universe, he will find scarcely anything which fails to please him in some way by its presence”*

**Marcus Aurelius**



## Acknowledgements

The book that you hold in your hands is a result of four years of intensive work, which consumed much of my time, also after the regular office hours. The thesis you are about to read has been forged in the fire of heated academic discussions; cooled later by careful lab work and data analysis. During this process I interacted with, and was influenced by, so many people that to name them all in one place is simply impossible. Therefore, I will restrict myself to naming only those who have had the strongest influence on my doctoral research.

I have to start by noticing that the research presented in this book is a part of a larger project aiming at developing intelligent dredgers. This ambitious research was initiated by the IHC Systems and the completion of this dissertation would not be possible without the financial and the logistic support of the company.

I would like to thank my supervisors from TU Delft: Robert Babuska, Zsófia Lendek, and Arjan den Dekker, and from the IHC Systems: Cees de Keizer and Jelmer Braaksma. Robert, Zsófia, and Arjan, thank you for teaching me how to operate in the academic world and how to become an independent researcher. Cees and Jelmer, thank you for keeping my mind focused on real-life problems and for making me understand the value of the Ockham razor. I collectively thank all of you for teaching me how to neatly formulate my thoughts and put them on paper. It was a great pleasure to interact with each one of you and I hope that our professional paths will keep crossing in the future.

Special thanks to prof. Prashant Mehta for hosting me for three months at the Coordinated Science Laboratory, at the UIUC. It was a great experience and a very refreshing one to be a part of your lab.

Great thanks to Alessandro for many stimulating discussions and for helping me to develop my teaching skills with only limited damage to my students.

I wish to thank Edwin. Firstly, for being an ambitious MSc student who challenged my intellect during many fruitful discussions, secondly, for being a great colleague later on.

I want to thank my committee members for precious feedback that helped improve the initial draft of my thesis.

When I became a part of the DCSC four years ago I had no idea what a vibrant and multicultural environment I would be joining. Interacting with people with so many different backgrounds often forced me to rethink my positions thus leaving a significant mark on my research. I thank Andrea for his stoic style that urges you to stop and always think twice before making any decision. I thank Alexandar for his limitless energy, which does not tolerate any melancholy in its presence. I thank Jacopo for his cynical attitude outmatched only by my own. I thank Ivo for being the first flying Dutchman whom I met. I thank Justin for shipping his enthusiasm from the hearth of the New World to the midst of phlegmatic Europe. I thank

Samira and Noor for showing me the crushing strength of the soft power. I thank Ilhan for his razor-sharp irony and for the optimal beard shaving theory. I thank Gabriel for showing me that the quietest person is the one that everybody listens to with the most attention. I thank Arturo for disapproving of the previous sentence with loud laughter. I thank Ilya for seeing the world through the same glasses that I do. I thank Patricio for his iron handshake as I thank Ania for her gentle manners. I thank Manuel for his lack of mercy when rebuffing my ill-formed arguments. And finally: what's up Marco?

I also want to thank my sister Asia who has been my proofreader over many years, Alex who has been my proofreader over many beers, and Ingmar, Edwin, and Pieter for their help with Dutch translations.

Thanks to Adam and Tao with whom I shared many interesting discussions during my stay at the UIUC.

# Contents

<b>Contents</b>	<b>vii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation	1
1.2 Research Goals	2
1.3 Outline of the Thesis	3
<b>2 Modeling and Estimation Problems in the Hopper Dredger</b>	<b>7</b>
2.1 Modeling of the Hopper Dredger	7
2.2 Drag-Head Excavation Model	8
2.3 Drag-Head Estimation Problems	12
2.3.1 Cutting Estimation Problem	13
2.3.2 Cutting and Jetting Estimation Problem	15
2.4 Hopper Sedimentation Model	16
2.5 Hopper Estimation Problems	23
2.6 Concluding Remarks	27
<b>3 Nonlinear Bayesian Filtering</b>	<b>29</b>
3.1 Introduction	29
3.2 Bayesian Dynamic Filtering	30
3.2.1 Bayesian Filter	30
3.2.2 Performance Evaluation	33
3.2.3 Optimal Filter for Linear Systems	35
3.3 Parametric Nonlinear Bayesian Filtering	38
3.3.1 Analytical Approximations	38
3.3.2 Statistical Approximations	43
3.3.3 Gaussian Sum Filter	52
3.4 Nonparametric Nonlinear Bayesian Filtering	60
3.4.1 Generic Particle Filter	60
3.4.2 Sequential Importance Sampling	61
3.4.3 Mean-Field Control-Oriented Approach	64
3.5 Conclusions and Discussion	66
<b>4 Solutions to the Drag-Head Estimation Problems</b>	<b>69</b>
4.1 Introduction	69
4.2 Handling the Time-Varying Measurement Delay	70
4.2.1 Measurement Delay in the Drag-Head System	70
4.2.2 Measurement Synchronization	71

## CONTENTS

---

4.2.3	Implementation of the Drag-Head Estimation . . . . .	74
4.3	Solutions to the Cutting Estimation Problem . . . . .	74
4.3.1	Derivation of the Discrete-Time Stochastic System . . . . .	76
4.3.2	Numerical Simulations . . . . .	78
4.3.3	Discussion . . . . .	82
4.4	Solutions to the Cutting and Jetting Estimation Problem . . . . .	84
4.4.1	Formulation of the Discrete-Time Stochastic System . . . . .	84
4.4.2	Numerical Simulations . . . . .	88
4.4.3	Discussion . . . . .	92
4.5	Conclusions . . . . .	93
<b>5</b>	<b>Saturated Particle Filter</b> . . . . .	<b>95</b>
5.1	Introduction . . . . .	95
5.1.1	Motivation . . . . .	96
5.1.2	Generic Saturated Stochastic Dynamical System . . . . .	96
5.2	One-Dimensional Saturated Particle Filter . . . . .	97
5.2.1	One-Dimensional Saturated Stochastic Dynamical System . . . . .	97
5.2.2	One-Dimensional Saturated Particle Filter . . . . .	99
5.2.3	Numerical Simulations . . . . .	102
5.3	Convex Saturated Particle Filter . . . . .	105
5.3.1	Convex Saturated Stochastic Dynamical System . . . . .	106
5.3.2	Convex Saturated Particle Filter . . . . .	108
5.3.3	Numerical Simulations . . . . .	112
5.4	Conclusions and Discussion . . . . .	116
<b>6</b>	<b>Asymptotic Properties of the Saturated Particle Filter</b> . . . . .	<b>121</b>
6.1	Introduction . . . . .	121
6.2	Preliminaries . . . . .	122
6.3	Asymptotic Properties of the SPF Under Standard Resampling . . . . .	124
6.3.1	Theoretical Results . . . . .	124
6.3.2	Practical Considerations . . . . .	127
6.4	Asymptotic Behavior of the SPF Under Improved Resampling . . . . .	127
6.4.1	Motivation . . . . .	128
6.4.2	New Resampling . . . . .	128
6.4.3	Almost Sure Convergence . . . . .	129
6.5	Properties of the Improved Saturated Particle Filter . . . . .	132
6.5.1	Implementation . . . . .	132
6.5.2	Detection Function . . . . .	134
6.5.3	Numerical Example . . . . .	136
6.6	Conclusions . . . . .	137
<b>7</b>	<b>Solution to the Hopper Estimation Problem</b> . . . . .	<b>141</b>
7.1	Introduction . . . . .	141
7.2	Overflow Loading Phases with Weak Erosion: the Reduced-Order Particle Filter . . . . .	142
7.2.1	Derivation of the Discrete-Time Stochastic System . . . . .	143
7.2.2	Numerical Simulations . . . . .	146
7.2.3	Discussion . . . . .	150
7.3	No-Overflow Loading Phase: the Feedback Particle Filter . . . . .	150
7.3.1	Derivation of the Stochastic System . . . . .	151
7.3.2	Numerical Simulations . . . . .	152

7.3.3 Discussion . . . . .	156
7.4 Overflow Loading Phases with Strong Erosion: the Improved Saturated Particle Filter . . . . .	159
7.4.1 Derivation of the Stochastic Dynamical System . . . . .	159
7.4.2 Algorithmic Properties . . . . .	161
7.4.3 Numerical Simulations . . . . .	162
7.4.4 Discussion . . . . .	164
7.5 Conclusions . . . . .	165
<b>8 Conclusions</b>	<b>167</b>
8.1 Summary . . . . .	167
8.2 Thesis Contributions . . . . .	169
8.2.1 Theoretical Contributions . . . . .	169
8.2.2 Practical Contributions . . . . .	170
8.3 Further research . . . . .	171
<b>A Uncertainty Analysis via Monte Carlo Simulations</b>	<b>173</b>
<b>B Proof of Proposition 7.1</b>	<b>179</b>
<b>Nomenclature</b>	<b>182</b>
<b>References</b>	<b>197</b>

## CONTENTS

---

# Chapter 1

## Introduction

### 1.1 Motivation

The main task of a *Trailing Suction Hopper Dredger* (TSHD) is to excavate sediments from the sea bottom while sailing and to transport them to a designated area. Its mobility and efficiency makes the TSHD an indispensable machine for large-scale land reclamation projects. That is why TSHDs are met all over the world, and were used during projects such as Chek Lap Kok airport in Hong Kong, Maasvlakte in Rotterdam port, Singapore port, the Palm Islands and The World Islands in Dubai, to name a few.

The dredging cycle of the TSHD starts when the ship sails off to the designated area. After the destination is reached, suitable in situ material is excavated from the bottom with a tool called the drag-head and it is automatically transported through a pipe to a cargo-hold where it is temporarily stored. In the hopper, the sediment settles at the bottom of the tank while the excess water is discharged overboard. After the hopper is full, the dredging is completed. The TSHD sails to the specified location where the collected material is discharged. The unloading is done either by opening the bottom doors of the ship, by rainbowing, or by pumping the material out of the hopper by the dredge pumps.

Currently TSHD operations are controlled by one or more operators. Consequently, the performance and efficiency of the entire process heavily depend on the insight and experience of the operators. For further improvement in efficiency the automation of the TSHD and the optimization of its performance is of crucial importance for dredging companies. Such tasks call for an integral approach that takes into account the overall dredging cycle, as well as the separate processes such as the excavation process, the sedimentation process, the pipeline transportation process, the discharge process, etc. As the automation of the operating system of the TSHD has been a subject of intensive studies, over the last decades, a rich literature dealing with the TSHD-related processes has accumulated [[Braaksma et al., 2007a,b,c](#); [Hahlbrock and Freese, 1998](#); [Ikeda et al., 1995](#); [Kurita et al., 1992](#); [Matoušek, 2002](#); [Miedema, 1984](#); [Morita et al., 2002](#); [Ooijens, 1999](#); [van Rhee, 2002a,b](#)]. The call for an automated controller of the TSHD has been answered in [[Braaksma, 2008](#)] where a *Model Predictive Controller* (MPC) for the TSHD has been developed.

The dynamics of the processes that occur onboard the TSHD, e.g., the excavation process or the sedimentation process, heavily depend on the properties of the in situ soil. Consequently, the performance of a controller that uses the models of these processes is strongly dependent on the detailed knowledge of the in situ soil. As the soil conditions vary continuously throughout the dredging cycle, it is, in general, impossible to obtain such a knowledge beforehand.

## 1. INTRODUCTION

---

Therefore, the properties of the soil have to be determined online, during the dredging, from the measurements available onboard. Although modern dredgers are equipped with advanced sensors measuring various variables needed by the controller, no direct measurements of the soil-dependent parameters can be obtained. Therefore, the soil properties have to be determined from the indirect observations of the states of the system.

The time-varying nature of the soil-dependent parameters combined with the model uncertainty and inaccuracy in the measurements make the estimation a challenging problem which is the main motivation behind the research presented in this thesis.

### 1.2 Research Goals

The main objective of the research discussed in this thesis is to recover certain properties of the in situ soil from the measurements available on board the TSHD. Namely, we are interested in estimating soil-dependent parameters and states of the most crucial processes controlled by the aforementioned MPC. Due to the time-varying, highly uncertain and complex nature of these processes we are interested in the estimators that are adaptable to changing conditions and can deal with the stochasticity of the signals. Moreover, the estimators have to be computationally efficient as the estimates need to be fed online to the automatic controller of the TSHD.

The soil-dependent parameters that are the most important for the controller of the TSHD are the horizontal cutting force coefficient  $k_{ch}$ , the ratio  $k_{vh}$  between the horizontal and vertical cutting forces, the in situ permeability  $k_{si}$ , and the average grain diameter  $d_m$ . These parameters are essential for the control of the drag-head excavation process and the hopper sedimentation process.

The drag-head is a part of the excavation system of the TSHD. It breaks the coherence of the in situ soil, which is next sucked up by the dredge pump and transported through the pipeline into the hopper. To accurately control the drag-head three soil-dependent parameters need to be estimated: the ratio  $k_{vh}$  between cutting forces, the horizontal cutting force coefficient  $k_{ch}$ , and the in situ permeability  $k_{si}$ .

The hopper sedimentation process describes the settling of the material transported through the pipeline into the tank. The settling rate of the material strongly depends on the type of soil that was pumped into the hopper. More precisely, the sedimentation depends on the average grain diameter  $d_m$  of the excavated soil. The accurate knowledge of  $d_m$  is necessary to control the sedimentation process in an optimal way, i.e., to maximize the production of the TSHD, given the hard constraints such as the maximum volume of the hopper or the maximum weight of the TSHD.

The main objective of this research is to find solutions to the estimation problems associated with two aforementioned processes. Namely, we want to:

1. *Solve the Drag-Head Estimation Problem*

design an estimator for:

- the ratio  $k_{vh}$  between cutting forces,
- the horizontal cutting force coefficient  $k_{ch}$ ,
- the in situ permeability  $k_{si}$ .

2. *Solve the Hopper Estimation Problem*

- design an estimator for the average grain diameter  $d_m$  of the excavated soil.

The estimators that solve the above estimation problems have to be:

1. *Accurate and precise*: to maximize the performance of the controller that uses the estimates,
2. *Numerically efficient*: to be feasible for online applications,
3. *Adaptive*: to adapt to continuously changing environment during the dredging operations,
4. *Robust*: to handle strong uncertainties in the models,
5. *Nonlinear*: to handle severe nonlinearities in the models.

Thus, we focus our investigations on the Nonlinear Bayesian Filters, which have all these properties.

First, we review established filtering algorithms satisfying the aforementioned properties in search for a method that provides the best solution to the drag-head and Hopper Estimation Problems. To evaluate their performance, we perform multiple numerical simulations of the dredging operations. When standard methods do not produce satisfactory results, we develop novel filtering algorithms that take advantage of the specific structures of the excavation model and of the sedimentation model.

## 1.3 Outline of the Thesis

The thesis is composed of two parts, a theoretical part and an application part. The application part spans over Chapters 2, 4, and 7. The theoretical part is composed of Chapters 3, 5, and 6. The diagram showing the relations between the two parts as well as between the individual chapters is presented in Figure 1.1. In order to understand Chapter 4 one should first read Chapters 2 and 3. Before reading Chapter 7 it is recommended to read Chapters 2, 3 and 6. To follow the theoretical developments presented in the thesis it is recommended to read Section 3.4, Chapter 5 and Chapter 6 in this order.

Chapter 2 serves as an introduction to dynamical modeling of the TSHD. Therein we present the dynamical models of the drag-head excavation process and the hopper sedimentation process and we formulate the corresponding estimation problems that are solved in further chapters.

Chapter 3 reviews Bayesian filters. We distinguish two types of methods: parametric methods and nonparametric methods. Among the parametric methods we consider three types of filters: filters based on analytical approximations (Extended Kalman Filter, Iterated Extended Kalman Filter), filters based on statistical approximations (Unscented Kalman Filter, Central Difference Filter, Gauss-Hermite Filter), and filters based on the Gaussian Sum Approximation (Gaussian Sum Filter). Among the nonparametric methods we review the Monte Carlo algorithms based on the importance sampling approach (Bootstrap Particle Filter) and based on the mean-field control-oriented approach (Feedback Particle Filter).

In Chapter 4 we use several of the methods reviewed in Chapter 3 to solve the Drag-Head Estimation Problems introduced in Chapter 2. We discuss the solutions to two estimation problems associated with the excavation process: the Cutting Estimation Problem that comes from considering the cutting-only production mode, and the Cutting and Jetting Estimation Problem that originates from the complete cutting and jetting production mode. Beside the estimation problems we develop a method of handling the time-varying delay in measurements used by the drag-head excavation model.

In Chapter 5, a novel filtering method is derived for stochastic dynamical systems with some state variables being constrained or saturated. We show how the estimates obtained by the existing methods can be improved by incorporating the measurements into the filtering step

## 1. INTRODUCTION

---



Figure 1.1: Structure of the thesis. The arrows indicate the dependencies between the different chapters.

of the algorithm through a user-specified detection function, which aims to detect the saturation as it occurs. We derive the Saturated Particle Filter (SPF) for a class of systems with one-dimensional constraints and we further extend our approach to multidimensional systems with convex constraints, deriving the Convex Saturated Particle Filter (CSPF). The effectiveness of the proposed methods is illustrated on examples which show that both the SPF and the CSPF achieve high accuracy using relatively few particles, thus preserving the low computational complexity of the algorithm.

In Chapter 6 we investigate the asymptotic properties of the filter developed in Chapter 5, in particular its almost sure convergence to the true posterior PDF. Furthermore, an improved SPF

is developed that uses a novel resampling procedure to overcome the practical shortcomings of the original SPF. We prove that this new filter also converges almost surely to the true posterior PDF.

In Chapter 7 we use several of the nonparametric methods reviewed in Chapter 3 as well as the methods developed in Chapters 5 and 6 to solve the Hopper Estimation Problem introduced in Chapter 2. The final solution to the Hopper Estimation Problem is obtained by integrating the filters designed for the separate modes, which appear naturally during the dredging operations, into a global estimator.

Chapter 8 concludes the thesis.

## 1. INTRODUCTION

---

## Chapter 2

# Modeling and Estimation Problems in the Hopper Dredger

### Abstract

A Trailing Suction Hopper Dredger (TSHD) is a ship used in various excavation projects. Its main task is to collect soil from a dredging zone and transport it to a designated area. Due to enormous scale of modern dredging operations the optimization of the whole dredging process is of crucial importance for dredging industry. One of the main problem that is faced by operators of TSHDs is that the implementation of efficient dredging strategies depends on the detailed knowledge of the in situ soil. Unfortunately, in general, such knowledge is difficult to obtain. In fact, soil-dependent parameters which are crucial to control the most important components of the dredging cycle cannot be directly measured onboard modern TSHDs and need to be estimated instead.

In this chapter we present dynamical models of the drag-head excavation process and the hopper sedimentation process. Each of these models contains soil-dependent parameters that need to be estimated for control purposes. These are: the horizontal cutting force coefficient  $k_{ch}$ , the ratio  $k_{vh}$  between the horizontal and vertical cutting forces, the in situ permeability  $k_{si}$  (drag-head model), and the average grain diameter  $d_m$  (hopper model). For each of these models we formulate the corresponding estimation problems that will be solved in further chapters.

### 2.1 Modeling of the Hopper Dredger

The main purpose of a *Trailing Suction Hopper Dredger* (TSHD) is to excavate sediments from the sea or river bottom while sailing. First, the in situ material is excavated with a tool called the drag-head, then it is hydraulically transported through a pipe to a cargo-hold (the hopper) where it is temporarily stored. After the operation is terminated, the ship sails to the designated site where the collected material is discharged.

The optimization of dredging operations is of vital importance for future reduction of costs in terms of time, labor, and resources. While modern hopper dredgers are equipped with advanced dynamic positioning and tracking systems, no on-board decision-support systems are yet available to optimize the dredging performance under given operating conditions (type of soil, dredging depth, water current, etc.). The manipulated variables must constantly be adjusted by one or two operators: the ship navigator and the dredge process operator. Consequently, the performance and efficiency of the entire process heavily depend on their insight and experience.

## 2. MODELING AND ESTIMATION PROBLEMS IN THE HOPPER DREDGER

---



Figure 2.1: TSHD Marieke during dredging operations in the North Sea (courtesy of Jelmer Braaksma).

In recent years the automation of the operation system of TSHD has been an intensively studied subject in the dredging community. In the literature, models of the isolated components of the total system were developed [Braaksma et al., 2007b; Ikeda et al., 1995; Matoušek, 2002; Miedema, 1984; Ooijens, 1999; van Rhee, 2002a,b], together with the overall model taking into account the interactions between separate subsystems [Braaksma et al., 2007a,b,c].

In this chapter we describe two of the most important parts of the TSHD model:

1. the drag-head model which describes the excavation process of the material which is further transported to the hopper,
2. the hopper model which, among others, describes the sedimentation of the material excavated from the bottom.

We present dynamical state space models for both aforementioned processes. Furthermore, we formulate estimation problems that result from a number of uncertain parameters which are used in the modeling process.

Finally, we have to mention that the overall dynamical model of the TSHD consists of more submodels, one of which also contains soil-dependent parameters. This is the Pump-Pipeline model [Braaksma, 2008] and the parameters are: the transport factor  $S_{kt}$  and the Stepanoff correlation  $\gamma$ . Sensors currently available onboard are not sufficient to obtain estimates of these parameters by considering the Pump-Pipeline system as a standalone block. However, it is possible to describe the soil-dependent parameters  $S_{kt}$  and  $\gamma$  as functions of the average grain diameter  $d_m$  [Braaksma, 2008] which appears in the Hopper Sedimentation Model. This means that accurate estimates of  $d_m$  can be further used by the Pump-Pipeline model. This concept has not been tested in practice and we leave it as a topic for future research.

### 2.2 Drag-Head Excavation Model

The drag-head is the most important component of the excavation system in a hopper dredger. Its task is to break the coherence of the bottom soil which allows the loosened material to be sucked in by the dredge pump and be transported to the storage tank (hopper). The overall production comes from three factors: production by erosion, production by the water jets and production by cutting. Among these three, the first component, i.e., the production by erosion is considered as a factor of a negligible influence and is not considered in the model. Recently

## 2.2. Drag-Head Excavation Model

it has been argued that although the erosion factor is in general of the lowest importance, in some cases the production obtained by erosion is a significant part of the overall production. Nevertheless, at this moment we have no model for this process, hence we assume that the overall production comes only from cutting and jetting.

The water jets are placed in the heel of the drag-head (see Figure 2.3). They are powered by an onboard jet pump which supplies the energy to jet the water under high pressure into the bottom. This loosens the material and allows it to be sucked by the dredge pump into the transport pipe. It has been claimed that the water jets are the most effective when dredging compacted fine sand.

The cutting device is placed on the other side of the drag-head (see Figure 2.3). It consists of a blade or several teeth which cut through the soil making it possible to be transported to the hopper. The necessary energy is supplied by the propulsion of the ship. The cutting production is the most effective when the soil is hard packed sand.



Figure 2.2: A drag-head (courtesy of IHC Systems).

There are three distinct operating modes that describe the excavation process. They are characterized by the relation between the excavation depth  $h_{ex}$  and the visor depth  $h_v$ . The drag-head is most effective when the excavation depth  $h_{ex}$  is equal to the visor depth  $h_v$  ( $h_{ex} = h_v$ ). When this is the case there are no gaps between the bottom and the heel ( $h_{ex} > h_v$ ) nor between the bottom and the teeth ( $h_{ex} < h_v$ ). As a result, a high jetting production rate is combined with a high cutting production rate. This regime is schematically presented in Figure 2.3.

The other two excavation regimes refer to situations when there is a gap between the teeth and the bottom, i.e.,  $h_{ex} > h_v$  or between the heel and the bottom, i.e.,  $h_{ex} < h_v$ . In the first case no cutting takes place which means that the entire production is obtained solely through jetting. In the second case the highest possible cutting production is achieved, but the jetting production is reduced due to a low pressure drop over the drag-head. Throughout this thesis we shall not consider these two regimes but we restrict our analysis only to the no-gap excavation regime.

The volume balance of the drag-head excavation process is given by

$$Q_m = Q_{s,j} + Q_{s,c} + Q_{w,j} + Q_{w,t} + Q_{w,v}, \quad (2.1)$$

where on the left hand side we have the production mixture flow  $Q_m$  and on the right hand side we can distinguish two types of flows:

1. flow of the material (sand) loosened by the teeth  $Q_{s,c}$  and by the jets  $Q_{s,j}$ ,

## 2. MODELING AND ESTIMATION PROBLEMS IN THE HOPPER DREDGER

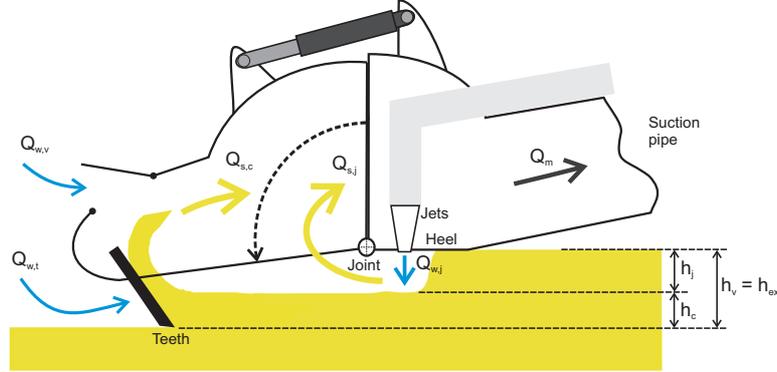


Figure 2.3: Schematic view of the drag-head with the important flows during the no-gap excavation regime (courtesy of Peter van den Bergh).

- the flow of water produced by jets  $Q_{w,j}$ , flow of water from surroundings  $Q_{w,t}$ , flow of water through the valve  $Q_{w,v}$ .

The production comes only from the flows  $Q_{s,c}$  and  $Q_{s,j}$  which are the main concern of the remainder of the section. The sand flows can be described in terms of jetting depth  $h_j$ , cutting depth  $h_c$ , the speed of the ship  $v_{sh}$  and the width of the drag-head  $W_d$  by the following formulas:

$$Q_{s,c} = h_c v_{sh} W_d, \quad (2.2a)$$

$$Q_{s,j} = h_j v_{sh} W_d. \quad (2.2b)$$

In the above formulation  $W_d$  is a known parameter and  $v_{sh}$  is measured on board of the ship. The models for the jetting and the cutting depths are discussed in detail in what follows.

### Cutting Model

The cutting depth  $h_c$  is calculated from the equilibrium of moments around the visor joint J. The schematic picture of a cutting tool (see Figure 2.4) with the corresponding geometrical scheme (see Figure 2.5) yields the following relation between the visor force  $F_{vc}$ , the horizontal cutting force  $F_{ch}$  and the vertical cutting force  $F_{cv}$ :

$$F_{vc} = \frac{x_2}{r_{vc}} (F_{ch} \sin(\alpha_{lt} + \alpha_v) + (F_{cv} \cos(\alpha_{lt} + \alpha_v))), \quad (2.3)$$

where the visor length  $x_2$  is a known parameter depending on the geometry of the drag-head. The moment arm  $r_{vc}$ , the visor angle  $\alpha_v$  and the angle of the lower suction pipe  $\alpha_{lt}$  are measured variables.

We neglect the friction force component because it is much smaller than the cutting force. Then, the motion of the cutting tool is described by the following differential equation:

$$\dot{v}_{sh} = \frac{1}{m_t} (F_{th} - F_{ch}) \quad (2.4)$$

where  $m_t$  is the total mass of the ship,  $F_{th}$  is the thrust force of the propeller blades and  $F_{ch}$  denotes the horizontal cutting force.

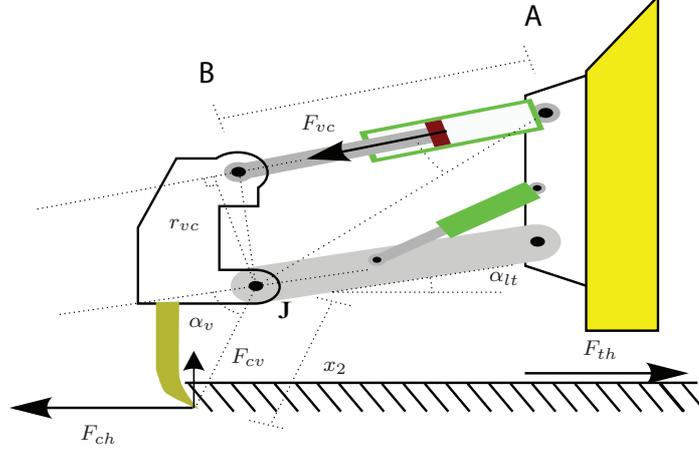


Figure 2.4: Schematic representation of a cutting tool (courtesy of Jelmer Braaksma).

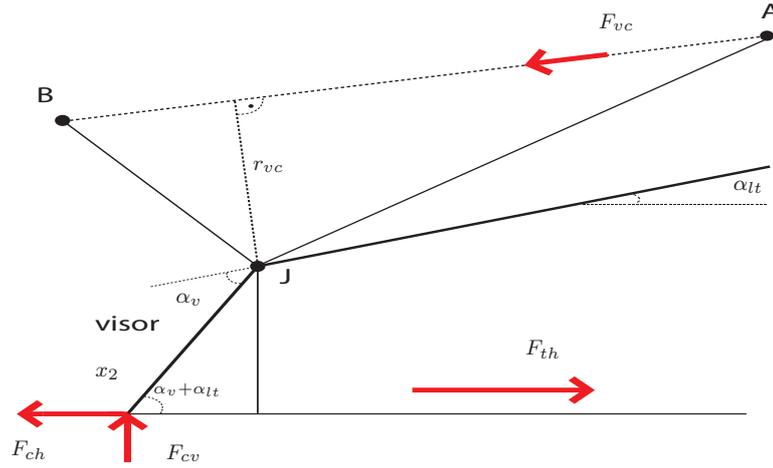


Figure 2.5: Geometry of the cutting tool. Thick arrows indicate the acting forces (courtesy of Peter van den Bergh).

From Miedema cutting theory [Braaksma, 2008; Miedema, 1987, 1996], assuming cavitating cutting, the cutting force  $F_{vc}$  is decomposed into a horizontal part  $F_{ch}$  and a vertical part  $F_{cv}$  from which we get the following relations between the cutting depth and the cutting forces:

$$F_{ch} = k_{ch} h_c (h_z + 10), \quad (2.5a)$$

$$F_{cv} = k_{cv} h_c (h_z + 10), \quad (2.5b)$$

where the dredging depth  $h_z$  is a measured variable and the cutting force coefficients  $k_{ch}$  and  $k_{cv}$  are unknown parameters that depend on the in situ soil. Because the parameters  $k_{ch}$  and  $k_{cv}$  are determined by the type of soil that is excavated, the values of these variables change when the drag-head encounters different soil. Therefore, these parameters are dynamic.

We assume that for each soil type the relation between the horizontal and vertical cutting

## 2. MODELING AND ESTIMATION PROBLEMS IN THE HOPPER DREDGER

---

force is fixed, i.e.,:

$$F_{cv} = k_{vh}F_{ch}, \quad (2.6)$$

where  $k_{vh}$  is the ratio between horizontal cutting force coefficient  $k_{ch}$  and vertical cutting force coefficient  $k_{cv}$ :

$$k_{vh} = \frac{k_{cv}}{k_{ch}}. \quad (2.7)$$

Then we can write

$$F_{cv} = k_{vh}F_{ch} = k_{vh}k_{ch}h_c(h_z + 10). \quad (2.8)$$

Combining (2.8) with (2.5a) and (2.3) yields

$$F_{vc} = \frac{x_2}{r_{vc}}h_c k_{ch}(\sin(\alpha_{lt} + \alpha_v) + k_{vh} \cos(\alpha_{lt} + \alpha_v))(h_z + 10), \quad (2.9)$$

from which we obtain the formula for the cutting depth

$$h_c = \frac{r_{vc}}{x_2}F_{vc}k_{ch}^{-1}(\sin(\alpha_{lt} + \alpha_v) + k_{vh} \cos(\alpha_{lt} + \alpha_v))^{-1}(h_z + 10)^{-1}. \quad (2.10)$$

With regard to the ratio  $k_{vh}$  we consider two possibilities:

1. the ratio  $k_{vh}$  is a constant parameter for a given soil type and varies between different soils,
2. the ratio  $k_{vh}$  is constant, regardless of the soil type.

In the first case  $k_{vh}$  becomes a dynamic parameter that changes as the excavated soil changes, whereas in the second case the  $k_{vh}$  is a static parameter that can be calibrated offline.

### Jetting Model

Based on empirical studies and from [Vlasblom, 2003] the model of jetting depth  $h_j$  is:

$$h_j := C_{dh}p_j^{0.5}Q_{w,j}v_{sh}^{-1}k_{si}^p, \quad (2.11)$$

where the constant  $C_{dh}$  is a known parameter dependent on the type of the drag-head used,  $p$  is a known fractional power determined from the experimental data, the jet nozzle pressure  $p_j$ , the jet water flow  $Q_{w,j}$  and the speed of the ship  $v_{sh}$  can be calculated from measurements. The only unknown in (2.11) is the in situ permeability  $k_{si}$  that depends on the type of the excavated material.

### 2.3 Drag-Head Estimation Problems

In this section we introduce a mathematical description of estimation problems related to the drag-head model. These problems arise due to the presence of uncertain soil dependent parameters such as the ratio between cutting forces  $k_{vh}$ , the horizontal cutting force coefficient  $k_{ch}$  and in situ permeability  $k_{si}$ .

We start by listing the drag-head related measurements available on board TSHD.

Table 2.1: Precisions of the available measurements

Symbol	Physical meaning	Precision	Unit
$p_j$	jet nozzle pressure	$\pm 7500$	[Pa]
$Q_{w,j}$	the jet water flow	$\pm 0.012$	[m <sup>3</sup> /s]
$v_{sh}$	speed of the ship	$\pm 0.2$	[m/s]
$r_{vc}$	the moment arm	$\pm 0.003$	[m]
$F_{vc}$	visor cylinder force	$\pm 13000$	[N]
$\alpha_{lt}$	angle of the lower suction pipe	$\pm 0.004$	[rad]
$\alpha_v$	visor angle	$\pm 0.005$	[rad]
$h_z$	dredging depth	$\pm 0.17$	[m]

### Onboard Sensors and Measurements

On board of the TSHD there are several sensors capable of taking online estimates that can be used in the drag-head model. The measured variables with benchmark precisions are reported in Table 2.1.

We assume that the values in the third column of Table 2.1 define a ball around the true value of the signal that contains 98 percent of the observations. Due to lack of further knowledge on the precision of the measurements or the biases involved we assume that the precision variable reported in Table 2.1 corresponds to  $3\sigma$  of the Gaussian distribution centered around the true value of the measurements (i.e., we assume that the measurements are unbiased).

### Delay in the Measurements of $h_{ex}$

The production mixture flow  $Q_m$  together with the mixture density  $\rho_m^{dh}$  are measured by the sensors located after the dredge pump. This means that the flow sensor and the density sensor are located far from the drag-head, in some cases up to 60 meters away. Thus, these crucial measurements that are used in mass-volume balance equations are delayed. Such a transport delay is dynamic and depends on the average flow rate and the length of the pipeline. The presence of the delay poses extra challenge in both control and estimation.

In the dynamical system that models the excavation process, the measurement of the excavation depth  $h_{ex}$  is assumed to be available. However,  $h_{ex}$  is calculated from the values of the incoming flow rate  $Q_i$ , incoming density (at the drag-head inlet)  $\rho_i^{dh}$ , the in situ sand density  $\rho_s$ , the ship's speed  $v_{sh}$ , the fixed values of water density  $\rho_w$  and the width of the drag-head  $W_d$  by the formula:

$$h_{ex}(t) = \frac{Q_i(t) (\rho_i^{dh}(t) - \rho_w)}{(\rho_{si}(t) - \rho_w) W_d v_{sh}(t)}. \quad (2.12)$$

The signals  $Q_i(t)$ ,  $\rho_{si}(t)$  and  $v_{sh}(t)$  are assumed to be known at time  $t$  without errors (thus, we treat them as deterministic inputs). The value of the incoming density at the drag-head inlet  $\rho_i^{dh}$  is measured with the transport delay  $\tau_t$  at the pump  $\rho_i^m$ . Hence, we have the relation:

$$\rho_i^m(t) = \rho_i^{dh}(t - \tau_t). \quad (2.13)$$

It should be noted that the delay  $\tau$  cannot be calculated forward in time but only backward.

#### 2.3.1 Cutting Estimation Problem

First we formulate an estimation problem for the drag-head not employing jets during the production process. In such a case the total production comes only from the cutting tool

## 2. MODELING AND ESTIMATION PROBLEMS IN THE HOPPER DREDGER

---

schematically depicted in Figure 2.4 which means that the excavation depth  $h_{ex}$  is equal to the cutting depth  $h_c$  ( $h_j = 0$ ).

The inputs to the system are defined in what follows.

### Input Signals

The thrust force  $F_{th}$  can be calculated from the propellers' shaft speeds and the pitches of the propellers which are controllable variables. Thus, we can also assume that the thrust force  $F_{th}$  is a known input to the system. Recall that in the absence of jetting production  $h_{ex} = h_c$  which means that in the cutting-only regime the cutting depth  $h_c$  is measured.

For the Cutting Estimation Problem we assume that the uncertainties in the measurements of the excavation depth  $h_{ex}$ , the visor angle  $\alpha_v$ , the angle of the lower suction pipe  $\alpha_{lt}$ , the moment arm  $r_{vc}$  and the total mass  $m_t$  are negligible. This means that we treat all these variables as known inputs to the system.

### The Estimation Objective

The cutting model described in the previous section contains two uncertain parameters which depend on the in situ soil. During the dredging operation while the ship sails the value of  $k_{ch}$  changes. Furthermore, we assume that the ratio  $k_{vh}$  also varies as the type of the excavated soil changes. Therefore, to model the dynamical nature of these soil-dependent parameters,  $k_{ch}$  and  $k_{vh}$  are modeled as time-varying parameters.

Due to the lack of a mathematical model that would describe the evolution of  $k_{ch}$  or  $k_{vh}$  we use the random walk approach to describe the evolution of (unmodeled) uncertain parameters [Ionides et al., 2006; Kitagawa, 1998]. Thus, we have modeled the dynamics of  $k_{vh}$  and  $k_{ch}$  with the zero-drift stochastic differential equations:

$$dk_{vh}(t) = 0dt + de_{vh}(t), \quad (2.14a)$$

$$dk_{ch}(t) = 0dt + de_{ch}(t), \quad (2.14b)$$

where  $e_{vh}$  and  $e_{ch}$  are Wiener processes with constant standard deviations  $\sigma_{vh}$  and  $\sigma_{ch}$ , respectively.

Then, the cutting-only model is given by the stochastic differential equations:

$$dv_{sh}(t) = \frac{1}{m_t(t)} (F_{th}(t) - F_{ch}(t)) dt + de_{vsh}, \quad (2.15a)$$

$$dk_{vh}(t) = 0dt + de_{vh}(t), \quad (2.15b)$$

$$dk_{ch}(t) = 0dt + de_{ch}(t), \quad (2.15c)$$

$$F_{vc} = \frac{x_2}{r_{vc}} k_{ch} h_c (\sin(\alpha_{lt} + \alpha_v) + k_{vh} \cos(\alpha_{lt} + \alpha_v)). \quad (2.15d)$$

where to the deterministic parts developed in previous section we added the stochastic components  $de_{vsh}, de_{vh}, de_{ch}$  to model the uncertainty in variables  $v_{sh}, k_{vh}$  and  $k_{ch}$ .

We assume that two variables are measured: the speed of the ship  $v_{sh}$  and the visor cylinder force  $F_{vc}$ . The measurements are assumed to be corrupted by zero-mean, time-invariant Gaussian noises  $e_t^{vsh}$  and  $e_s^{Fvc}$ , respectively.

Given the system (2.15) with the known inputs and uncertain observations we formulate the estimation problem as obtaining online estimates of the uncertain soil-dependent parameters:

1. the ratio  $k_{vh}$  between the cutting forces,

2. the horizontal cutting force coefficient  $k_{ch}$ ,

from the available measurements.

### 2.3.2 Cutting and Jetting Estimation Problem

Let us now formulate the estimation problem for the drag-head that combines the cutting production and the jetting production. We start by defining the input signals.

#### Input Signals

Assuming that the measurements listed in Table 2.1 are available, we define the following two signals:

$$u_c := \frac{r_{vc}}{x_2} F_{vc} (\sin(\alpha_{lt} + \alpha_v) + k_{vh} \cos(\alpha_{lt} + \alpha_v))^{-1} (h_z + 10)^{-1}, \quad (2.16a)$$

$$u_j := C_{dh} p_j^{0.5} Q_{w,j} v_{sh}^{-1}, \quad (2.16b)$$

where  $x_2$  and  $C_{dh}$  are parameters known from the specifications of the drag-head and  $k_{vh}$  defined in (2.7) is assumed to be known.

Let us discuss a setting, where both  $u_c$  and  $u_j$  are corrupted by noises that come from other measured variables ( $p_j, Q_{w,j}, v_{sh}, r_{vc}, F_{vc}, \alpha_{lt}, \alpha_v, h_z$ ) such that each of them carries an uncertainty in it. The precisions of these variables are given in Table 2.1.

For the proper and complete analysis of the uncertainty in the signals  $u_j$  and  $u_c$ , which are derived from the signals in the table by (2.16a)–(2.16b), the distribution of  $u_j$  and  $u_c$  as functions of Gaussians should be derived. However, since these functions are severely nonlinear, such a theoretical analysis leads to complex distributions that are unfeasible for practical implementations. Thus, we rely on approximations. Namely, we decided to analyze these distributions by Monte Carlo experiments discussed in detail in Appendix A.

The simulations suggest that the noise associated with  $u_c$  is distributed normally with zero-mean, with the standard deviations being roughly the same for all the values of the variable  $u_c$  and equal to  $\sigma_{u_c} = 120.3$  (see Appendix A). The noise associated with  $u_j$  is slightly skewed to the right, which suggests the presence of a bias in the signal. Therefore, in this case a Gaussian approximation is less accurate. Furthermore, the standard deviation of such an approximation depends on the value of  $u_j$  and can be approximated by the following function<sup>1</sup>:

$$\sigma_{u_j} = 0.04874 + 0.04857u_j^{2.693}. \quad (2.17)$$

#### The Estimation Objective

With the use of (4.27) we rewrite (2.10)–(2.11) as:

$$h_c = u_c k_{ch}^{-1}, \quad (2.18a)$$

$$h_j = u_j k_{si}^p. \quad (2.18b)$$

Given that in the no-gap regime (see Figure 2.3) we have

$$h_{ex} = h_c + h_j, \quad (2.19)$$

---

<sup>1</sup>The approximation was obtained by using the MATLAB function *fit.m*.

## 2. MODELING AND ESTIMATION PROBLEMS IN THE HOPPER DREDGER

---

and the fact that the excavation depth  $h_{ex}$  is assumed to be measured (with a delay) the objective is to estimate the unknown soil-dependent parameters  $k_{ch}$  and  $k_{si}$  from the available signals  $u_c, u_j, h_{ex}$  and from the formula:

$$h_{ex} = u_c k_{ch}^{-1} + u_j k_{si}^p. \quad (2.20)$$

Apart from soil-dependent parameters: the cutting force coefficient  $k_{ch}$  and in situ permeability  $k_{si}$ , all the variables used in (2.20) are known or measured. The parameters  $k_{ch}$  and  $k_{si}$  are unknown to the operator and they change dynamically during the dredging operation when the type of the excavated material changes. Table 4.3 reports experimentally obtained values of both parameters for eight most common soil types. As the parameters  $k_{ch}$  and  $k_{si}$  depend on the type of soil excavated by the drag-head, they are correlated. However, unlike the Hopper Sedimentation model, there is no known way of establishing accurate relations between those parameters and a single soil-dependent parameter.

Table 2.2: The cutting force coefficient  $k_{ch}$  and the in situ permeability  $k_{si}$  for different soil types.

Soil type	Soil type A (Fine)		Soil type B (Medium)		Soil type C (Medium)		Soil type D (Coarse)	
	Medium	Dense	Medium	Dense	Medium	Dense	Medium	Dense
$d_m$ [mm]	0.10	0.10	0.24	0.24	0.45	0.45	1.30	1.30
$k_{ch}$	$9.87 \cdot 10^4$	$1.16 \cdot 10^5$	$9.56 \cdot 10^4$	$1.12 \cdot 10^5$	$9.53 \cdot 10^4$	$1.11 \cdot 10^5$	$8.89 \cdot 10^4$	$1.03 \cdot 10^5$
$k_{si}$ [m/s]	$3.59 \cdot 10^{-5}$	$1.6 \cdot 10^{-5}$	$2.75 \cdot 10^{-4}$	$1.3 \cdot 10^{-4}$	$4.06 \cdot 10^{-4}$	$1.81 \cdot 10^{-4}$	$2.86 \cdot 10^{-3}$	$1.23 \cdot 10^{-3}$

Note that if all the signals in (2.20) are stochastic and evolve independently of each other with the same frequency, the model is underdetermined and cannot be solved with respect to  $k_{ch}$  and  $k_{si}$ . Therefore, the main objective is to obtain online estimates of the uncertain soil-dependent parameters:

1. the cutting force coefficient  $k_{ch}$ ,
2. the in situ permeability  $k_{si}$ ,

from the collected measurements of  $p_j, Q_{w,j}, v_{sh}, r_{vc}, F_{vc}, \alpha_{lt}, \alpha_v, h_z$ , and  $h_{ex}$ .

### 2.4 Hopper Sedimentation Model

The sedimentation process has been extensively studied in the civil engineering literature [Camp, 1946; Felice, 1999; Mirza and Richardson, 1979; Ooijens et al., 2001; Richardson and Zaki, 1954; van Rhee, 2002a].

Existing dynamical models derived from the Navier-Stokes equations [van Rhee, 2002a,b] are very detailed descriptions of the physical phenomenon in terms of 1-D, or 2-D, spatial *Partial Differential Equations* (PDE). Furthermore, such models contain a large number of uncertain parameters corresponding to the environmental properties of the excavated material. Thus, the overall complexity of such models makes them unfeasible for onboard online controllers.

A simplified 1-D sedimentation model was proposed in [Braaksma et al., 2007b] as a basis of the onboard controller of the dredging process. This model has been integrated with other models into a global *Model Predictive Controller* (MPC) of the TSHD [Braaksma, 2008]. Thus, in what follows we analyze the properties of the aforementioned 1-D sedimentation model. In particular, we formulate the estimation problems that need to be solved before the control can be applied.

The sand is the most common soil type excavated during the dredging operations, other soil types less frequently met during dredging being clay, silt or gravel. Therefore, in what follows we shall consider only the case when the in situ soil is known to be sand. We show how, under such an assumption, it is possible to reduce all the uncertain soil-related parameters to one parameter: the average grain diameter  $d_m$ .



Figure 2.6: Example of a hopper.

Before we give the equations that define the dynamical system describing the sedimentation process we shall briefly discuss the production cycle in a TSHD.

### The Production Process

The production process in a TSHD with retractable overflow system is naturally divided into three separate phases:

1. *The no-overflow phase.*
2. *The constant-volume phase.*
3. *The constant-tonnage phase.*

When the ship arrives at the dredging area, the loading begins. At first (no-overflow phase) all the excavated material is stored in the hopper. When the mixture level reaches a certain height, the second phase begins (constant-volume phase). During this stage the excess water (or a low density mixture) is being discharged overboard to keep the volume  $V_t$  of the stored material constant. As a result the density of the remaining mixture increases and therefore the total mass  $m_t$  of the material in the hopper also increases. The last loading phase begins after the maximum allowed mass in the hopper (determined by the maximum draught of the ship) has been reached. In order to prevent the ship from sinking a constant-tonnage controller is used. When necessary, the controller lowers the overflow height hence more mixture is disposed through the overflow pipe.

During this third phase the overflow losses increase up to the point when it is no longer economically efficient to continue dredging, at which point the loading stops.

## 2. MODELING AND ESTIMATION PROBLEMS IN THE HOPPER DREDGER

---

Sedimentation during each of the loading phases is characterized by different dynamics. As the process progresses from lower to higher loading phases the dynamics of the sedimentation models become increasingly complex [Braaksma et al., 2007b]. However, from the qualitative point of view we distinguish only two modes:

1. no-overflow mode,
2. the overflow mode.

The former is characterized by low erosion effect on the sedimentation, thus, as we shall see, simpler dynamics. The more challenging dynamics of the second mode are compensated by the higher number of observed outputs.

### Conservation Laws

The dynamics of the sedimentation process in the hopper during the no-overflow phase are derived from the conservation laws (mass balance, volume balance):

$$\dot{m}_s = Q_s(d_m, m_t, h_s, h_t, Q_o) \rho_s(d_m), \quad (2.21a)$$

$$\dot{V}_t = Q_i - Q_o, \quad (2.21b)$$

$$\dot{m}_t = Q_i \rho_i - Q_o \rho_o, \quad (2.21c)$$

where  $m_s$  denotes the mass of the sand bed and the total volume and total mass of the mixture in the hopper are denoted by  $V_t$  and  $m_t$ , respectively. The incoming flow rate is given by  $Q_i$ , and the density of the incoming mixture is denoted by  $\rho_i$ . The total height and the sand bed height are denoted by  $h_t$  and  $h_s$ , respectively. The settling sand flow rate  $Q_s$  and the density of the settled sand  $\rho_s$  are modeled as functions dependent on the average grain diameter  $d_m$ .

The overflow rate  $Q_o$  and the overflow density  $\rho_o$  are output variables which cannot be directly measured due to the lack of appropriate sensors in the overflow system. In the literature, a number of models of the overflow rate  $Q_o$  and the overflow density  $\rho_o$  have been proposed [Braaksma et al., 2007b]. Unfortunately, those models contain too many uncertain parameters which lead to rather inaccurate approximations of the desired signals, when compared with the measured data. Therefore, a cascaded observer of the overflow rate  $Q_o$  and the overflow density  $\rho_o$  has been developed in [Lendek et al., 2008] in order to obtain accurate online estimates of both signals. Thus, in this paper, the two aforementioned variables are regarded as measured outputs of the system.

The cross section of the hopper is visualized in Figure 2.7.

The settling sand flow rate  $Q_s$ , which appears in (2.21a), is modeled as a function of five parameters: the average grain diameter  $d_m$ , sand bed mass  $m_s$ , the total height of the mixture in the hopper  $h_t$ , sand bed height  $h_s$ , the overflow rate  $Q_o$ , and the total mass in the hopper  $m_t$  [Braaksma et al., 2007b].

The nonlinear function  $Q_s$  is factorized into two components

$$Q_s(d_m, m_t, h_s, h_t, Q_o) = f_e(d_m, h_t, h_s, Q_o) f_s(d_m, m_t, h_s, h_t), \quad (2.22)$$

where each term describes a different physical phenomenon. The scouring function  $f_e$  models the settling efficiency influenced by the erosion, which depends on the local mixture flow above the settled material [van Rhee, 2002b]:

$$f_e(d_m, h_t, h_s, Q_o) = \max \left( 1 - \frac{Q_o^2}{(k_e(d_m)(h_t - h_s))^2}, 0 \right), \quad (2.23)$$

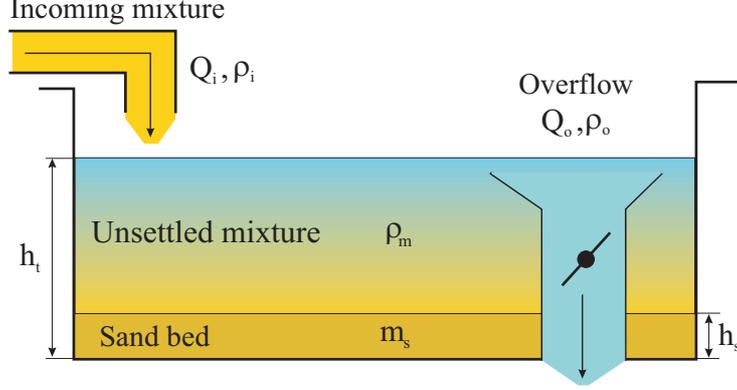


Figure 2.7: Cross section of the hopper showing the variables that are used to describe the sedimentation process.

where the erosion pickup flux coefficient  $k_e$  is a soil dependent parameter expressed as a function of the mean grain diameter  $d_m$  of the in situ material. The right-hand side of (2.23) takes values between 0 and 1 hence, by (2.22), the settling rate  $Q_s$  can only be decreased by the scouring function  $f_e$ . This corresponds to the negative effect of erosion on the sedimentation process.

The settling function  $f_s$  describes the process of settling of the sand particles suspended in the mixture above the sand bed. For a hopper of a rectangular parallelepiped shape with a base area  $A$  [m<sup>2</sup>]  $f_s$  is given by:

$$f_s(d_m, m_t, h_s, h_t) = Av_{s0}(d_m) \frac{\rho_m(m_t, h_s, d_m, h_t) - \rho_w}{\rho_s(d_m) - \rho_m(m_t, h_s, d_m, h_t)} \left( \frac{\rho_q - \rho_m(m_t, h_s, d_m, h_t)}{\rho_q - \rho_w} \right)^{\beta(d_m)}, \quad (2.24)$$

where  $\rho_w$  is the density of water (1024 [kg/m<sup>3</sup>]) and  $\rho_q$  is the density of quartz (approximately 2650 [kg/m<sup>3</sup>]). In (2.23) and (2.24) there are three soil dependent parameters described as functions of the average grain diameter  $d_m$ . These are: the sand bed density in the hopper  $\rho_s$ , the undisturbed settling velocity of a single particle  $v_{s0}$ , and the Richardson-Zaki exponent  $\beta$  [Richardson and Zaki, 1954].

Finally, the settling sand flow rate  $Q_s$  also depends on the density of the mixture in the hopper. It has been experimentally shown [Ooijens, 1999; van Rhee, 2002b] that above the sand bed, the mixture of water and sand that is being discharged into the hopper form a uniformly dense soup with a thin layer of water on the top. Thus, the density of the mixture can be approximated by the average density of the mixture  $\rho_m$ , given by [Braaksma et al., 2007b]:

$$\rho_m(m_t, h_s, d_m, h_t) = \frac{m_t - Ah_s\rho_s(d_m)}{Ah_t - Ah_s}. \quad (2.25)$$

To derive the dynamic model for the sand bed height  $h_s$  let us note that the increments of the sand bed mass  $m_s$  can be written in terms of the increments of the sand bed height  $h_s$  and the average of the sand bed density  $\rho_s$ :

$$m_s((k+1)T_s) - m_s(kT_s) = A(h_s((k+1)T_s) - h_s(kT_s)) \frac{1}{T_s} \int_{kT_s}^{(k+1)T_s} \rho_s(d_m) dt. \quad (2.26)$$

Thus, taking the limit  $T_s \rightarrow 0$ , we obtain the continuous-time ODE description of  $m_s$ :

$$\dot{m}_s = Ah_s\rho_s(d_m), \quad (2.27)$$

## 2. MODELING AND ESTIMATION PROBLEMS IN THE HOPPER DREDGER

---

from which it is straightforward to obtain the formula for the sand bed height  $h_s$  growth rate:

$$\dot{h}_s = \frac{Q_s(d_m, m_s, h_t, h_s, Q_o, m_t)}{A\rho_s(d_m)}. \quad (2.28)$$

The dynamics of the total height of the mixture  $h_t$  are derived from the difference between the incoming flow  $Q_i$  and the outgoing flow  $Q_o$  divided by the hopper area  $A$ :

$$\dot{h}_t = \frac{Q_i - Q_o}{A}. \quad (2.29)$$

### Soil-Dependent Parameters as Functions of $d_m$

The simplified 1-D sedimentation model contains four uncertain parameters: the undisturbed settling velocity  $v_{s0}$ , the sand bed density  $\rho_s$ , the Richardson-Zaki exponent  $\beta$ , and the erosion coefficient  $k_e$ . These depend only on the properties of the excavated soil. The successful implementation of the MPC controller requires these parameters to be specified. To complete the derivation of the sedimentation dynamics we need to specify the aforementioned soil-dependent parameters as functions of the average grain diameter  $d_m$ . Some of these functions are given explicitly in the literature, others have been estimated by least squares fit to the experimental data.

The relations between the sand bed density in the hopper  $\rho_s$  and  $d_m$  are established using the experimental results reported in [Braaksma, 2008]. These are given in Table 2.3.

Table 2.3: Empirical relations between  $\rho_s$  and  $d_m$

$d_m$ [mm]	0.09	0.12	0.19	0.30	0.86
$\rho_s$ [kg/m <sup>3</sup> ]	1934	1938	1942	1947	1957

The data reported in Table 2.3 gives only a rough approximation of the true functional relation between  $d_m$  and  $\rho_s$ . However, due to lack of other measurements they need to suffice. The curve fitting results in the following formula for  $\rho_s$ :

$$\rho_s(d_m) = 1926 + 34.81\sqrt{d_m}. \quad (2.30)$$

By [Matoušek, 1997] the undisturbed settling velocity  $v_{s0}$  is derived, for three distinctive regimes, from Stokes, Budryck or Rittinger equations:

$$v_{s0}(d_m) = \begin{cases} 424 \frac{\rho_q - \rho_w}{\rho_w} d_m^2 & d_m < 0.1 \text{ [mm] (Stokes)} & (2.31a) \\ \frac{8.925}{d_m} \left( \sqrt{1 + 95 \frac{\rho_q - \rho_w}{\rho_w} d_m^3} - 1 \right) & 0.1 < d_m < 1 \text{ [mm] (Budryck)} & (2.31b) \\ 87 \sqrt{\frac{\rho_q - \rho_w}{\rho_w} d_m} & d_m > 1 \text{ [mm] (Rittinger)} & (2.31c) \end{cases}$$

Throughout this thesis we consider the average grain diameter  $d_m$  to take values in the interval  $[0.1, 1]$ [mm], thus only the Budryck equation is of interest. The relation between  $d_m$  and  $v_{s0}$  is illustrated in Figure 2.9b.

According to [Camp, 1946; Vlasblom and Miedema, 1995] the erosion coefficient  $k_e$  as a function of  $d_m$  is derived from:

$$k_e(d_m) = W_{sh} \sqrt{\frac{8(1-n)\mu g}{f} \frac{\rho_q - \rho_w}{\rho_w} d_m}, \quad (2.32)$$

## 2.4. Hopper Sedimentation Model

where  $W_{sh}$  is the hopper width,  $f$  is the friction force coefficient,  $\mu$  is a coefficient dependent on the internal friction of the sediment,  $n$  is the porosity of the sand bed and  $g$  stands for the gravitational acceleration. Unfortunately, it is not easy to evaluate the parameters of (2.32) for the whole spectrum of values of  $d_m$ . Therefore, we use the empirical data reported in [Braaksma, 2008] and presented in Table 2.4 to establish the formula for the erosion coefficient  $k_e$ :

$$k_e(d_m) = 28.06\sqrt{d_m} - 6.35. \quad (2.33)$$

Table 2.4: Empirical relations between  $k_e$  and  $d_m$

$d_m$ [mm]	0.09	0.12	0.19	0.30	0.86
$k_e$	2	4	6	8	20

The Richardson-Zaki exponent  $\beta$  can be represented as a function of the Reynolds number  $Re_p$  [Richardson and Zaki, 1954; Rowe, 1987; van Rhee, 2002b]:

$$\beta(Re_p) = \frac{4.7 + 0.41Re_p^{0.75}}{1 + 0.175Re_p^{0.75}}. \quad (2.34)$$

In order to describe  $\beta$  as function of the average grain diameter  $d_m$  it is sufficient to approximate  $Re_p$  as a function of  $d_m$ . This is done in what follows. From [Matoušek, 2001] we know that:

$$Re_p = \frac{v_{ts}}{v_f} d_m, \quad (2.35)$$

where  $v_{ts}$  is the terminal settling velocity of a solid particle and  $v_f$  stands for the kinematic viscosity of the fluid. The latter can be expressed as:

$$v_f = \frac{40}{20 + T} \left[ \frac{\text{mm}^2}{\text{s}} \right], \quad (2.36)$$

where  $T$  is the temperature of fluid in degrees Celsius [van Rhee, 2002b]. It is inconvenient to have  $v_f$  as a function of  $T$  due to high variation of this signal. Instead we take the average of  $v_f$  for temperatures in range  $[0, 26]$  to obtain the approximation  $v_f = 1.11$ .

The terminal settling velocity is given by [Matoušek, 2001]:

$$v_{ts} = \sqrt{\frac{4}{3} \frac{\rho_q - \rho_w}{\rho_w} \frac{g}{C_D} d_m}, \quad (2.37)$$

where  $C_D$  is the drag coefficient of flow round settling particle. The drag coefficient  $C_D$  is a function of  $Re_p$ :

$$C_D = \begin{cases} \frac{24}{Re_p} & d_m < 0.05[\text{mm}], & (2.38a) \\ \frac{24}{Re_p} (1 + 0.173Re_p^{0.657}) + \frac{0.413}{1 + 1.63 \cdot 10^4 Re_p^{-1.09}} & 0.05[\text{mm}] < d_m < 2[\text{mm}], & (2.38b) \\ 0.445 & d_m > 2[\text{mm}]. & (2.38c) \end{cases}$$

Since we are interested only in  $d_m$  that takes values in the interval  $[0.1, 1][\text{mm}]$  we analyze (2.38b). Combining (2.35)–(2.37) we find the relation between  $Re_p$  and  $d_m$ :

$$Re_p \sqrt{\frac{24}{Re_p} (1 + 0.173Re_p^{0.657}) + \frac{0.413}{1 + 1.63 \cdot 10^4 Re_p^{-1.09}}} = \sqrt{\frac{\frac{4}{3} \frac{\rho_q - \rho_w}{\rho_w} g}{v_f^2}} d_m^{\frac{3}{2}}. \quad (2.39)$$

## 2. MODELING AND ESTIMATION PROBLEMS IN THE HOPPER DREDGER

---

Unfortunately, from (2.39) it is impossible to obtain a closed analytical formula for the Reynolds number as a function of  $d_m$ . Therefore, we derive an approximation based on the empirical data from [Matoušek, 2001] and reported in Table 2.5:

$$Re_p(d_m) = -2.289 + 41.53d_m + 118.6d_m^2. \quad (2.40)$$

This is illustrated in Figure 2.8.

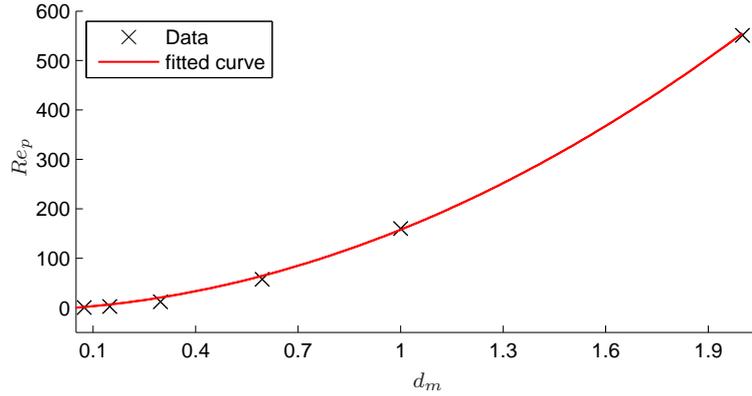


Figure 2.8: Approximation of the Reynolds number  $Re_p$  as a function of the average grain diameter  $d_m$ .

Table 2.5: Empirical relations between  $Re_p$  and  $d_m$

$d_m$ [mm]	0.074	0.149	0.297	0.595	1	2
$Re_p$	0.35	2.32	12.1	57.4	160	551

Combining (2.34)–(2.40) we get the formula for the Richardson-Zaki exponent for the average grain diameter  $0.1[\text{mm}] \leq d_m \leq 1[\text{mm}]$ :

$$\beta(d_m) = \frac{4.7 + 0.41 \cdot (-2.289 + 41.53d_m + 118.6d_m^2)^{0.75}}{1 + 0.175 \cdot (-2.289 + 41.53d_m + 118.6d_m^2)^{0.75}}. \quad (2.41)$$

To summarize, for a given hopper area  $A$ , the four soil dependent parameters  $\rho_s$ ,  $v_{s0}$ ,  $\beta$  and  $k_e$  can be approximated by the following functions of  $d_m$  [Braaksmas et al., 2007b; Rowe, 1987; Vlasblom and Miedema, 1995]:

$$\rho_s(d_m) = 34.81\sqrt{d_m} + 1926 \quad (2.42a)$$

$$v_{s0}(d_m) = \frac{8.925}{d_m} \left( \sqrt{1 + 95 \frac{\rho_q - \rho_w}{\rho_w} (d_m)^3} - 1 \right) \quad (2.42b)$$

$$\beta(d_m) = \frac{4.7 + 0.41 \left( -2.289 + 41.53d_m + 118.6 (d_m)^2 \right)^{0.75}}{1 + 0.175 \left( -2.289 + 41.53d_m + 118.6 (d_m)^2 \right)^{0.75}} \quad (2.42c)$$

$$k_e(d_m) = 28.06\sqrt{d_m} - 6.35. \quad (2.42d)$$

These functions are visualized in Figure 2.9.

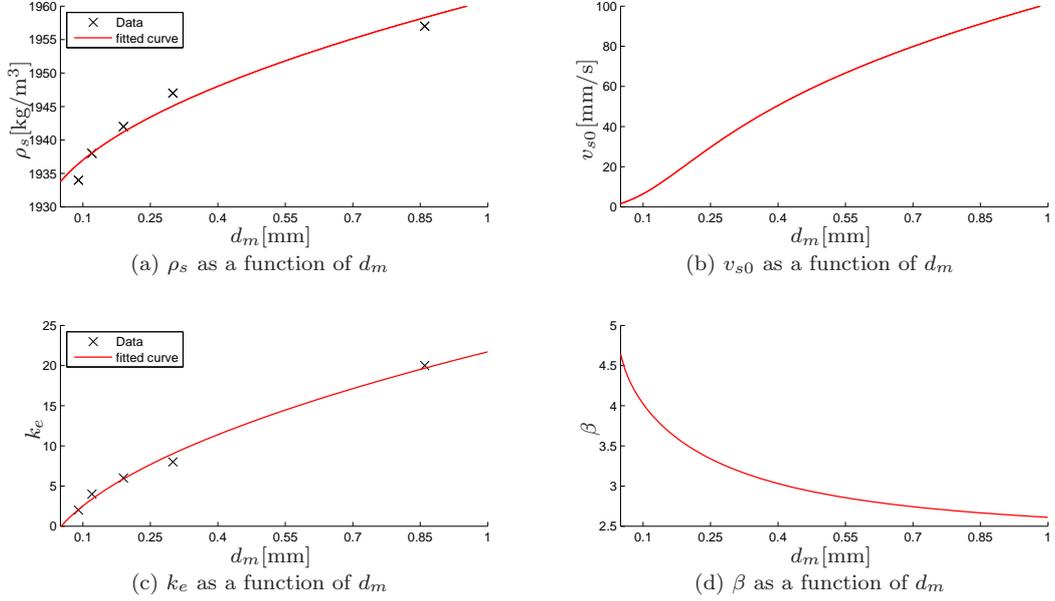


Figure 2.9: Approximation of the sand bed density in the hopper  $\rho_s$ , the undisturbed settling velocity  $v_{s0}$  (Budryck regime), the erosion coefficient  $k_e$  and the Richardson-Zaki exponent  $\beta$  as functions of the average grain diameter  $d_m$ .

### Dynamical model for the average grain diameter $d_m$

In the previous section we have shown that the dynamics of the 1-D sedimentation process can be derived from the conservation laws (2.21). The resulting dynamical system contains a number of parameters, which depend on the in situ soil. We have shown that these parameters can be approximated as functions of a single soil-dependent parameter: the average grain diameter  $d_m$ . During the dredging operation the ship is constantly sailing, and hence the value of  $d_m$  is subjected to changes as the type of the excavated soil changes. Therefore, to account for the temporal changes in the environment,  $d_m$  is modeled as a time-varying parameter. No mathematical model exists that would describe the evolution of the average grain diameter  $d_m$ . This poses problems for classical dynamical filters, which require a model for each estimated state. It has been argued [Ionides et al., 2006; Kitagawa, 1998] that for estimation purposes the evolution of (unmodeled) uncertain parameters can be described by a random walk. Thus, we have modeled the dynamics of the  $d_m$  with the zero-drift stochastic differential equation:

$$dd_m(t) = 0dt + de_d(t), \quad (2.43)$$

where  $e_d$  is a Wiener process with a constant standard deviation  $\sigma_d$ .

## 2.5 Hopper Estimation Problems

As was previously mentioned when the excavated soil is known to be sand the number of uncertain parameters can be reduced from four to one. This is possible because the aforementioned sedimentation parameters can be approximated [Braaksma et al., 2007b; Richardson and Zaki,

## 2. MODELING AND ESTIMATION PROBLEMS IN THE HOPPER DREDGER

---

1954; Rowe, 1987; Vlasblom and Miedema, 1995] by explicit functions of the average grain diameter of the excavated soil  $d_m$ . Thus, from the estimate of the  $d_m$ , it is possible to retrace the desired parameters of the simplified sedimentation model. Furthermore, the knowledge of the  $d_m$  can be used to estimate the uncertain soil-dependent parameters in other subsystems of the automated TSHD. For all the above reasons, development of an accurate estimator of the average grain diameter  $d_m$  is a crucial step towards the fully automated control system of the TSHD.

### Onboard Sensors and Measurements

On board of the TSHD there are several sensors capable of taking online measurements, which are further used in the Hopper Sedimentation model. Among them we distinguish five types:

1. Overflow height sensors,
2. Hopper mixture level sensors,
3. Pressure sensors in the bottom of the hull,
4. Radioactive density sensor placed in the discharged pipe,
5. Electromagnetic Flow meters.



Figure 2.10: Sensors of the overflow height (left) and the mixture level (right).

The sampling time is different for each sensor and for each ship and varies from 20[Hz] to 0.2[Hz].

The first sensor measures the height of the overflow weir  $h_o$  (Figure 2.10a) that is used by the constant-tonnage controller. The hopper mixture level sensors (Figure 2.10b) are used to measure the total height of the mixture in the hopper  $h_t$  (thus also the total volume  $V_t$ ). The pressure sensors in the bottom of the hull are used to calculate the draught of the ship. From the draught it is possible to calculate the mass of the ship. Thus the total mass of the mixture  $m_t$  is computed by subtracting the mass of an empty ship from the current mass of the TSHD. The radioactive source (Figure 2.11b) is used to measure the density of the incoming flow  $\rho_i$ . It is placed in the discharge pipe, directly above the pump. Finally, the Electromagnetic Flow meters are used to measure the velocity of the incoming mixture. Knowing the velocity and

the diameter of the transport pipe we can calculate the flow of the incoming mixture  $Q_i$ . For detailed descriptions of the sensors onboard the TSHD the reader is referred to [Braaksma, 2008].

Currently the TSHDs are not equipped with sensors that measure the sand bed height  $h_s$ . However, designing of such a sensor is in the advanced stage of development and it is believed that on future generations of ships it will be possible to measure  $h_s$ . That is why throughout this thesis we assume that the sand bed height  $h_s$  is measured.



Figure 2.11: Overflow weir (left) and the radioactive source in the discharged pipe (right).

### Estimated Signals

According to (2.21) the evolution of both the total mass  $m_t$  and the total volume  $V_t$  is determined by the incoming flow rate  $Q_i$ , and the density of the incoming mixture  $\rho_i$  (the no-overflow period) together with the outgoing flow rate  $Q_o$ , and the overflow density  $\rho_o$  (the constant-volume phase and the constant-tonnage phase). The first two signals  $Q_i$ , and  $\rho_i$  are measured in the discharge pipe (see Section 2.4.4 of [Braaksma, 2008]). The last two  $Q_o$ , and  $\rho_o$  are estimated online by an external cascaded estimator [Lendek et al., 2008] with the accuracy and precision given in Table 2.6

Table 2.6: Statistics of the residuals  $\tilde{\rho}_o$  and  $\tilde{Q}_o$

residual	mean	standard deviation	unit
$\tilde{\rho}_o$	10.679	21.54	[kg/m <sup>3</sup> ]
$\tilde{Q}_o$	0	0.6	[m <sup>3</sup> /s]

As it can be seen in Table 2.6 the estimates of the overflow density  $\hat{\rho}_o$  and the outgoing flow rate  $\hat{Q}_o$  closely match the true values of the signals  $\rho_o$  and  $Q_o$ <sup>1</sup>. Thus, for the purposes of online estimations, we can treat the signals  $\hat{\rho}_o$  and  $\hat{Q}_o$  as inputs to the model (2.21) used to estimate the average grain diameter  $d_m$ . This can be done in two possible ways:

- the estimates  $\hat{\rho}_o$  and  $\hat{Q}_o$  are considered to be deterministic inputs to the system,

<sup>1</sup>The typical values of the overflow density  $\rho_o$  vary from 1000 to 1500 [kg/m<sup>3</sup>], and the typical values of the outgoing flow  $Q_o$  vary from 7.5 to 11 [m<sup>3</sup>/s]

## 2. MODELING AND ESTIMATION PROBLEMS IN THE HOPPER DREDGER

---

- the estimates  $\hat{\rho}_o$  and  $\hat{Q}_o$  are considered to be stochastic inputs to the system, i.e., both signals are assumed to be corrupted by the Gaussian noises with means and standard deviations given in Table 2.6.

The former approach leads to a deterministic dynamical model for the  $m_t$  and  $V_t$  whereas the latter results in stochastic dynamical model for the aforementioned variables. Throughout this thesis we assume the former approach.

### Estimation Objective

In the previous section the 1-D dynamical sedimentation model has been derived. The model is based on the conservation laws augmented with the models for the soil-dependent parameters. The complete model is given by the stochastic differential equations:

$$dm_t(t) = (Q_i(t)\rho_i(t) - Q_o(t)\rho_o(t)) dt, \quad (2.44a)$$

$$dm_s(t) = Q_s(d_m(t), m_s(t), h_t(t), h_s(t), Q_o(t), m_t(t))dt + de_{m_s}(t), \quad (2.44b)$$

$$dh_s(t) = \frac{Q_s(d_m(t), m_s(t), h_t(t), h_s(t), Q_o(t), m_t(t))}{A\rho_s(d_m(t))}dt + de_s(t), \quad (2.44c)$$

$$dh_t(t) = \frac{Q_i(t) - Q_o(t)}{A}dt, \quad (2.44d)$$

$$dd_m(t) = 0dt + de_d(t), \quad (2.44e)$$

where to the deterministic parts developed in previous section we added the stochastic components  $de_{m_s}, de_s, de_d$  to model the uncertainty in variables  $m_s, h_s$  and  $d_m$ .

We assume that five variables are measured during the loading process: the height of the sand bed  $h_s$ , the total height of the mixture in the hopper  $h_t$ , the total mass of the mixture in the hopper  $m_t$ , the incoming flow rate  $Q_i$ , and the incoming flow  $\rho_i$ . The measurements are assumed to be corrupted by zero-mean, time-invariant Gaussian noises  $e_t^o, e_s^o, e_{m_t}^o, e_q^o$ , and  $e_i^o$ , respectively. Furthermore, two variables are assumed to be known exactly, i.e., without an error: the outgoing density  $\rho_o$  and the flow of the outgoing mixture  $Q_o$ .

Given the system (2.44) with the assumed measurements we formulate the primary estimation problem as obtaining online estimates of the uncertain soil-dependent parameter:

1. the average grain diameter  $d_m$ ,

from the available measurements. This is motivated by the fact that  $m_t$  and  $h_t$  can be accurately estimated directly from the available measurements of  $Q_i, \rho_i, Q_o$ , and  $\rho_o$ . Per contra the remaining variables, i.e.,  $m_s$ , and  $h_s$  cannot be accurately estimated without the accurate estimates of the  $d_m$ . Furthermore, the knowledge of the average grain diameter  $d_m$  gives a deep insight into the characteristics of the excavated soil which can be used to optimize the dredging operation.

The secondary objectives are the online estimation of the following variables:

2. the mass of the sand bed  $m_s$ ,
3. the sand bed height  $h_s$ ,
4. the density of the mixture  $\rho_m$ .

The variables  $m_s$ , and  $h_s$ , as it was noted above, can be accurately estimated given the accurate estimates of the  $d_m$ . The density of the mixture  $\rho_m$  is a function (2.25) of the state

variables  $m_t$ ,  $h_s$ ,  $d_m$ , and  $h_t$ . Therefore, given accurate estimates of the aforementioned variables we obtain accurate estimates of the  $\rho_m$ . Nevertheless, it is important to investigate the quality of the estimates of the variables  $m_s$ ,  $h_s$ , and  $\rho_m$  when accurate estimates of  $d_m$  are not available (due to e.g., a transient stage, or biased estimates).

## 2.6 Concluding Remarks

In this chapter we gave a general introduction to the operations of Hopper Dredgers. Furthermore, we have presented models of two main systems taken from the overall model of the TSHD. These are the drag-head Model and the Hopper Sedimentation Model for which we have formulated the estimation problems. In both systems the objective of the estimation is to retrieve the knowledge of the in-situ soil properties online from indirect measurements.

The uncertain and dynamical nature of the parameters of interest and the nonlinear dynamics of the considered systems in each case make estimation a challenging task. The algorithms that are suitable for these types of problems are reviewed in Chapter 3 and novel methods, tailored for specific types of systems, are developed in Chapter 5 and Chapter 6. Solutions to the estimation problems formulated in this chapter are provided in Chapter 4 (for the drag-head model) and in Chapter 7 (for the Hopper Sedimentation model).

## 2. MODELING AND ESTIMATION PROBLEMS IN THE HOPPER DREDGER

---

## Chapter 3

# Nonlinear Bayesian Filtering

Parts of this chapter were published in:

- “*Parametric Bayesian Filters for Nonlinear Stochastic Dynamical Systems: A Survey*”, Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews, Paweł Stano, Zsófia Lendek, Robert Babuška, Jelmer Braaksma, Cees de Keizer and Arnold J. den Dekker, in press.

### Abstract

Nonlinear stochastic dynamical systems are commonly used to model physical processes. For linear and Gaussian systems, the Kalman Filter is optimal in the minimum mean squared error sense. However, for nonlinear or non-Gaussian systems the estimation of states or parameters is a challenging problem. Furthermore, it is often required to process data online. Therefore, apart from being accurate, a feasible estimation algorithm also needs to be fast. In this chapter we review Bayesian filters which possess the aforementioned properties. Each filter is presented in an easy to implement algorithmic form. We focus on two types of filters: parametric methods and nonparametric methods. Among the parametric methods we distinguish three types of filters: filters based on analytical approximations (Extended Kalman Filter, Iterated Extended Kalman Filter), filters based on statistical approximations (Unscented Kalman Filter, Central Difference Filter, Gauss-Hermite Filter), and filters based on the Gaussian Sum Approximation (Gaussian Sum Filter). When discussing nonparametric methods we focus on Monte Carlo algorithms based on the importance sampling approach (Bootstrap Particle Filter) and based on the mean-field control-oriented approach (Feedback Particle Filter).

### 3.1 Introduction

The concept of *filtering* has been studied for decades in various engineering problems that require extracting information of interest from an uncertain or changing environment. A filter is a *recursive algorithm* designed for a case when the complete knowledge of the relevant signal characteristics is not available [Haykin, 1991]. The main purpose of a filter is to utilize the available information about the process of interest in order to obtain an estimate of certain variables that cannot be measured directly or precisely.

In this chapter we analyze filters designed for nonlinear discrete-time continuous-state dynamical systems. These are used to model, among others, physical [Weare, 2009], chemical [Murshed et al., 2010], biological [Barbieri et al., 2004], or economic [Danielsson, 1994]

### 3. NONLINEAR BAYESIAN FILTERING

---

processes. Usually, one is interested in the continuous-time phenomena, often governed by (partial) differential equations. However, due to the complexity of these models and the limited computational power available, simplifications are required in order to obtain an efficient solution. Discrete-time systems provide such a simplification since in this framework time is represented by the monotonic set of discrete time steps that allows the recursive filtering of the process of interest. Since the discretized system is only an approximation of the original one, there is always a certain degree of uncertainty incorporated into the model, which depends on the discretization technique that was applied [Rodríguez-Millán and González, 2005]. An other possible approximation is to replace the detailed deterministic (dynamical) relations with probabilistic approximations which, if appropriately chosen, further simplify the system. However, this comes with the price of increased uncertainty of the model (see [Talay, 1995] and references therein).

The main objective of this chapter is to review and discuss the filtering methods that are commonly applied to nonlinear stochastic dynamical systems. Among many techniques dealing with this subject we can distinguish: grid-based methods [Arulampalam et al., 2002; Bewley and Sharma, 2012; Ristic et al., 2004a] designed for dynamical systems defined on a finite state space, point-mass methods [Kramer and Sorenson, 1988; Šimandl et al., 2002, 2006] that are based on grid approximation of the continuous state space, Beneš-Daum filters [Beneš, 1981; Daum, 1986, 2005] derived for a specific class of nonlinear systems with linear observations, parametric methods [Ito and Xiong, 2000; Julier and Uhlmann, 2004a,b; Nørgaard et al., 2000], i.e., methods for which the estimation problem has a solution in a finite dimensional parameter space, nonparametric methods based on numerical integrations via Monte Carlo approach such as Particle Filters [Arulampalam et al., 2002; Blom and Bloem, 2011b; Cristian and Doucet, 2002; Doucet et al., 2000] or Ensemble Particle Filters [Burgers et al., 1998; Evensen, 2003, 2006; Gland et al., 2011] popular in data assimilation problems, and more [Arasaratnam and Haykin, 2009; Särkkä, 2012; Yang et al., 2011a]. Throughout the years, each of these approaches lead to the development of a multitude of algorithms. Detailed analysis of such a vast number of estimation techniques is a monumental task. Therefore, we focus our attention on parametric and nonparametric filters. We present filtering algorithms and investigate their properties and their feasibility for online applications.

## 3.2 Bayesian Dynamic Filtering

In this section we formulate the generic *Bayesian Filter* (BF) framework for nonlinear and non-Gaussian dynamical systems. First, let us define the probabilistic state-space system, which serves as a framework for the BF problem.

### 3.2.1 Bayesian Filter

#### General Framework

The probabilistic state-space framework relates the process model describing the evolution of the states in time, the observation model relating the noisy measurements of the system to the actual state, and the initial state of the system. In discrete-time, at each time instant  $k = 1, 2, \dots$ , the probabilistic state-space description is given by

*the state model*

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k, \mathbf{v}_k), \quad (3.1)$$

the observation model

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k, \mathbf{w}_k), \quad (3.2)$$

and the initial condition

$$\mathbf{x}_0 \sim p_0, \quad (3.3)$$

where  $\mathbf{x}_k \in \mathbb{R}^n$  and  $\mathbf{y}_k \in \mathbb{R}^p$  are random variables corresponding to the state model and the measurement model at time step  $k$ , respectively. Note that  $\mathbf{v}_k \in \mathbb{R}^d$  and  $\mathbf{w}_k \in \mathbb{R}^l$  are uncorrelated random variables, which represent the system noise and the measurement noise at time step  $k$ , and are independent of the distribution of the initial state  $p_0$ . Throughout the thesis we assume that  $\mathbf{f}_k : \mathbb{R}^n \times \mathbb{R}^d \rightarrow \mathbb{R}^n$  is a known nonlinear function that models the evolution of the state  $\mathbf{x}_k$  affected by the random variable  $\mathbf{v}_k$ , and that  $\mathbf{h}_k : \mathbb{R}^n \times \mathbb{R}^l \rightarrow \mathbb{R}^p$  is known nonlinear function that relates the observed variable  $\mathbf{y}_k$  to the state variable  $\mathbf{x}_k$  at time step  $k$  under the disturbances caused by the noise  $\mathbf{w}_k$ . Furthermore, we assume that the distributions of state and observation noise  $\mathbf{v}_k$  and  $\mathbf{w}_k$  are known for all  $k \geq 1$ .

Note that in some applications the functions  $\mathbf{f}_k$  or  $\mathbf{h}_k$  might depend on uncertain parameters [Roweis and Ghahramani, 2001]. In such cases it is possible to learn the dynamics of the system online with the Expectation-Maximization algorithm [Cappé and Moulines, 2009; Cappé et al., 2009; Roweis and Ghahramani, 2001].

Given the sequence of measurements up to time step  $k$ , i.e.,  $\mathcal{Y}_k = \{\mathbf{y}_i, i = 1, \dots, k\}$ , and the initial knowledge of the state distribution  $p_0$ , the objective of the estimation is to find a *probability density function* (PDF) of the state of the system. The PDF contains full information of the state variable from which it is possible to derive statistical properties of the state, such as mean, variance, etc. For dynamical systems we distinguish three classical estimation problems [Bar-Shalom et al., 2001; Kitagawa, 1987; Simon, 2006; Šimandl and Duník, 2009]:

1. *m-step smoothing*: estimation of  $p(\mathbf{x}_{k-m} | \mathcal{Y}_k)$ ,
2. *m-step prediction*: estimation of  $p(\mathbf{x}_{k+m} | \mathcal{Y}_k)$ ,
3. *filtering*: estimation of  $p(\mathbf{x}_k | \mathcal{Y}_k)$ .

One can distinguish three types of smoothing algorithms: fixed-point smoothers, which estimate a state at a fixed point of time using a growing number of measurements, fixed-interval smoothers, which estimate states within a fixed time interval using all the measurements from the same time interval, and fixed-lag smoothers, which estimate states with a fixed time delay. The overview of these methods is out of scope of this paper, instead readers interested in smoothing methods are referred to [Godsill et al., 2004; Kitagawa, 1987; Rauch et al., 1965; Särkkä, 2012; Särkkä and Hartikainen, 2010; Šimandl and Duník, 2009; Šimandl et al., 2001].

The prediction problem is closely related to the filtering problem. In fact, finding the  $m$ -step predictor can always be done by iterating the prediction step of a given filter. Thus, no specialized algorithms are needed for this.

In this paper we restrict our analysis to BFs, which recursively solve the filtering problem for the system (3.1)–(3.3) in two steps. First, during the *prediction step*, the state model (3.1) and the density  $p(\mathbf{x}_{k-1} | \mathcal{Y}_{k-1})$  are used to derive the *predicted state density* via the Chapman-Kolmogorov equation:

$$p(\mathbf{x}_k | \mathcal{Y}_{k-1}) = \int p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathcal{Y}_{k-1}) p(\mathbf{x}_{k-1} | \mathcal{Y}_{k-1}) d\mathbf{x}_{k-1} \quad (3.4a)$$

$$= \int p(\mathbf{x}_k | \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} | \mathcal{Y}_{k-1}) d\mathbf{x}_{k-1}. \quad (3.4b)$$

### 3. NONLINEAR BAYESIAN FILTERING

---

where the *transition density*  $p(\mathbf{x}_k|\mathbf{x}_{k-1})$  is determined by the known statistics of  $\mathbf{v}_{k-1}$  and the transformation  $\mathbf{f}_{k-1}$ . Note that (3.4b) follows by (3.1) which states that the state variable  $\mathbf{x}_k$  is conditionally independent of past measurements  $\mathcal{Y}_{k-1}$ .

The prediction step is followed by the *update step* where the most recent measurement  $\mathbf{y}_k$  is combined with the predicted state density  $p(\mathbf{x}_k|\mathcal{Y}_{k-1})$  using the observation model (3.2). The desired *posterior* PDF  $p(\mathbf{x}_k|\mathcal{Y}_k)$  is computed via Bayes' rule:

$$p(\mathbf{x}_k|\mathcal{Y}_k) = \frac{p(\mathbf{y}_k|\mathbf{x}_k, \mathcal{Y}_{k-1})p(\mathbf{x}_k|\mathcal{Y}_{k-1})}{p(\mathbf{y}_k|\mathcal{Y}_{k-1})} \quad (3.5a)$$

$$= \frac{p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathcal{Y}_{k-1})}{\int p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathcal{Y}_{k-1})d\mathbf{x}_k}. \quad (3.5b)$$

Equation (3.5b) is allowed because the observation model (3.2) assumes that the observation variable  $\mathbf{y}_k$  is conditionally independent of the past observations  $\mathcal{Y}_{k-1}$  given the state variable  $\mathbf{x}_k$ .

#### Outputs of the Estimator

Note that most real-life applications do not require a PDF but rather a concrete point estimate of a state. The posterior PDF contains all the information required for computing an optimal point estimate  $\hat{\mathbf{x}}_k$  of the state with respect to a predefined criterion. In general, the choice of the criterion is an important (and non trivial) problem. An incorrectly chosen  $\hat{\mathbf{x}}_k$  might lead to a significant decrease in the filter's performance as, e.g., in multi-target tracking applications [Aoki et al., 2011; Blom and Bloem, 2011a]. Two of the most popular estimators [Gauvain and Lee, 1994; Rangan et al., 2009; Ristic et al., 2004a] are the *minimum mean-square error* (MMSE) estimator and the *maximum a posteriori* (MAP) estimator. The MMSE estimate is computed as the conditional mean of  $\mathbf{x}_k$  given  $\mathcal{Y}_k$

$$\hat{\mathbf{x}}_k^{\text{MMSE}} = \mathbb{E}(\mathbf{x}_k|\mathcal{Y}_k) = \int \mathbf{x}_k p(\mathbf{x}_k|\mathcal{Y}_k) d\mathbf{x}_k.$$

The MAP estimate is given by the vector that maximizes the posterior density, i.e., it is a solution of the optimization problem

$$\hat{\mathbf{x}}_k^{\text{MAP}} = \arg \max_{\mathbf{x}_k} p(\mathbf{x}_k|\mathcal{Y}_k).$$

Note that the MAP estimate is not unique if the posterior PDF achieves the maximal value in multiple points (e.g., the PDF of the uniform distribution).

#### Suboptimality of Nonlinear Filters

For systems with linear dynamics and additive Gaussian noises [Ljung and Gunnarsson, 1990; Matthies et al., 1989; Øksendal, 2003b; Welch and Bishop, 1995] the posterior PDF is also Gaussian [Ho and Lee, 1964] and can be computed in a closed form by the *Kalman Filter* (KF) [Kalman, 1960]. The KF is an unbiased estimator [Anderson and Moore, 1979] that is optimal in the MMSE and in the MAP sense (for the KF the MMSE estimate and the MAP estimate are identical).

In case of nonlinear systems the optimal solutions to the filtering problem for dynamical systems defined on a finite state space are given by grid-based methods [Arulampalam et al., 2002; Bewley and Sharma, 2012; Ristic et al., 2004a]. Optimal Beneš-Daum filters [Beneš, 1981;

[Daum, 1986, 2005] have been derived for a specific class of nonlinear systems with linear observations. Unfortunately, for general nonlinear systems no closed form solution to the filtering problem exists. This is because the computation of the posterior densities (3.5) requires numerical integration of complicated, often high-dimensional, functions. Furthermore, for Bayesian filters the suboptimal performance can also be the result of imperfect knowledge of the initial distribution  $p_0$  [Kleptsyna and Veretennikov, 2011]. Thus, in most cases we need to rely on approximations of the true posterior PDF which lead to suboptimal solutions [Kushner and Dupuis, 2001; Pugachev and Sinitzyn, 2001].

In what follows we discuss several suboptimal filtering algorithms that employ different deterministic and probabilistic numerical methods used to approximate the integrals in (3.5b)–(3.4b). They differ by their computational complexity, and numerical accuracy. The decision regarding the “optimal” choice of the suboptimal algorithm is never straightforward and is influenced by factors such as: the dynamics of the system, the computational power available, observability of the model, uncertainty in parameters, etc.

### 3.2.2 Performance Evaluation

As was indicated in the previous section, for the general nonlinear filtering problem there are many suboptimal methods to choose from. In this section we show how to measure the performance of different filters.

#### Posterior Cramér Rao Bound

It is possible to assess the achievable performance of a given nonlinear filter by computing the (sampled) variance of the estimator and comparing it with the *Posterior Cramér-Rao Bound* (PCRB). The PCRB gives a lower bound on the *mean squared error* (MSE) for any estimator (see Chapter 2.4 of [van Trees, 1968]). Thus, it is a generalization of the classical *Cramér-Rao Bound* (CRB), see Chapter 32 of [Cramer, 1946] or [Rao, 1945], which bounds the MSE of estimators of deterministic variables. The PCRB is derived for the system (3.1)–(3.3) and it is independent of the filter applied to the system. Thus, the PCRB serves as a benchmark for comparing the performance of nonlinear filters [Farina et al., 2002]. Applications of CRB to continuous-time nonlinear filtering are discussed in [Kerr, 1989] whereas [Tichavsky et al., 1998] focuses on discrete-time nonlinear filtering.

In what follows  $\mathbf{x}_{1:k}$  and  $\mathbf{y}_{1:k}$  denote the random variables  $(\mathbf{x}_1, \dots, \mathbf{x}_k)$  and  $(\mathbf{y}_1, \dots, \mathbf{y}_k)$ , respectively.

The performance of a given estimator  $\hat{\mathbf{x}}_{1:k}$  is measured using the mean squared estimation error defined by

$$\mathbb{E}_{\mathbf{y}_{1:k}, \mathbf{x}_{1:k}} \left( (\hat{\mathbf{x}}_{1:k} - \mathbf{x}_{1:k}) (\hat{\mathbf{x}}_{1:k} - \mathbf{x}_{1:k})^T \right), \quad (3.6)$$

where  $\mathbb{E}_{\mathbf{y}_{1:k}, \mathbf{x}_{1:k}}$  denotes the expectation taken with respect to the random variables  $\mathbf{y}_{1:k}$  and  $\mathbf{x}_{1:k}$ , and  $\hat{\mathbf{x}}_{1:k}$  is an estimator of  $\mathbf{x}_{1:k}$  which depends on the observation  $\mathbf{y}_{1:k}$ .

The PCRB provides a lower bound on (3.9) which is given by the inverse of the information matrix  $(\mathbf{J}_{1:k})^{-1}$ , i.e.,

$$\mathbb{E}_{\mathbf{y}_{1:k}, \mathbf{x}_{1:k}} \left( (\hat{\mathbf{x}}_{1:k} - \mathbf{x}_{1:k}) (\hat{\mathbf{x}}_{1:k} - \mathbf{x}_{1:k})^T \right) \geq (\mathbf{J}_{1:k})^{-1}. \quad (3.7)$$

The  $nk \times nk$  information matrix  $\mathbf{J}_{1:k}$  is defined by [Tichavsky et al., 1998]

$$\mathbf{J}_{1:k} := \mathbb{E}_{\mathbf{y}_{1:k}, \mathbf{x}_{1:k}} \left( -\Delta_{\mathbf{x}_{1:k}}^{\mathbf{x}_{1:k}} \log p(\mathbf{x}_{1:k}, \mathbf{y}_{1:k}) \right), \quad (3.8)$$

### 3. NONLINEAR BAYESIAN FILTERING

---

where  $\Delta_x^y = \nabla_x (\nabla_y)^T$  is the second-order derivative and  $p(\mathbf{x}_{1:k}, \mathbf{y}_{1:k})$  is a joint density of the random variable  $(\mathbf{x}_{1:k}, \mathbf{y}_{1:k})$ .

Equations (3.7)–(3.8) give the lowest bound on the MSE of an estimator of the whole trajectory  $\hat{\mathbf{x}}_{1:k}$ . However, by (3.8), computation of the right-hand side of (3.7) requires inverting the large  $nk \times nk$  matrix  $\mathbf{J}_{1:k}$ , which is undesirable from the numerical perspective. Fortunately, it has been shown [Tichavsky et al., 1998] that it is possible to compute the PCRB recursively for each single-step estimator  $\hat{\mathbf{x}}_k$  as:

$$\mathbb{E}_{\mathbf{y}_{1:k}, \mathbf{x}_{1:k}} \left( (\hat{\mathbf{x}}_k - \mathbf{x}_k) (\hat{\mathbf{x}}_k - \mathbf{x}_k)^T \right) \geq (\mathbf{J}_k)^{-1}. \quad (3.9)$$

In (3.9),  $\mathbf{J}_k$  is a  $n \times n$  matrix that can be computed recursively by solving Riccati-like equations:

$$\mathbf{J}_{k+1} = \mathbf{D}_k^{22} - (\mathbf{D}_k^{12})^T (\mathbf{J}_k + \mathbf{D}_k^{11})^{-1} \mathbf{D}_k^{12}, \quad (3.10)$$

where

$$\mathbf{D}_k^{11} := \mathbb{E}_{\mathbf{x}_{1:k}} \left( -\Delta_{\mathbf{x}_k}^{\mathbf{x}_k} \log p(\mathbf{x}_{k+1} | \mathbf{x}_k) \right) \quad (3.11a)$$

$$\mathbf{D}_k^{12} := \mathbb{E}_{\mathbf{x}_{1:k}} \left( -\Delta_{\mathbf{x}_k}^{\mathbf{x}_{k+1}} \log p(\mathbf{x}_{k+1} | \mathbf{x}_k) \right) \quad (3.11b)$$

$$\mathbf{D}_k^{22} := \mathbb{E}_{\mathbf{x}_{1:k}} \left( -\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \log p(\mathbf{x}_{k+1} | \mathbf{x}_k) \right) + \mathbb{E}_{\mathbf{y}_{1:k}, \mathbf{x}_{1:k}} \left( -\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \log p(\mathbf{y}_{k+1} | \mathbf{x}_{k+1}) \right). \quad (3.11c)$$

The iteration (3.10) is initialized with matrix  $\mathbf{J}_0$ , which is calculated from the initial condition (3.3)

$$\mathbf{J}_0 := \mathbb{E}_{\mathbf{x}_0} \left( -\Delta_{\mathbf{x}_0}^{\mathbf{x}_0} \log p_0(\mathbf{x}_0) \right). \quad (3.12)$$

Thus, dealing with large matrices is avoided.

Note that the PCRB implementation requires the derivatives in (3.11) to be evaluated in the true states  $\mathbf{x}_k$  and  $\mathbf{x}_{k+1}$  [Farina et al., 2002; Lei et al., 2011]. Alternatively, the PCRB can be approximated by evaluating  $\mathbf{D}_k^{11}$ ,  $\mathbf{D}_k^{12}$  and  $\mathbf{D}_k^{22}$  in the estimate of the state [Lei et al., 2011]. It has been argued that such an approximated PCRB can also be used as a performance measure of nonlinear filters [Lei et al., 2011]. In some online applications the use of the *Conditional PCRB*, which depends on the actual realization  $\mathcal{Y}_k$  of the random variable  $\mathbf{y}_{1:k}$ , is preferable over the standard PCRB [Zuo et al., 2011]. Another interesting class of PCRBs used in target tracking applications and designed for systems with uncertainty about measurements origin has been studied in [Hernandez et al., 2004; Niu et al., 2001; Zhang et al., 2005]. Recursive algorithms for computing the PCRB for prediction, filtering and smoothing estimation problems are discussed in [Šimandl et al., 2001].

#### Stability of the Bayesian Filter

The PCRB is a very useful performance measure as it gives the precision of the optimal nonlinear filter and thus it also provides a limit on the performance achievable by any suboptimal method. Regarding the stability of the Bayesian filter we consider the following question: Let  $\pi_k = p(\mathbf{x}_k | \mathcal{Y}_k)$  and  $\tilde{\pi}_k = p(\mathbf{x}_k | \tilde{\mathcal{Y}}_k)$  be Bayesian filters defined by (3.4)–(3.5) starting from the initial distributions  $p_0$  and  $\tilde{p}_0$ , respectively. What is the difference between  $\pi_k$  and  $\tilde{\pi}_k$  as time progresses? Formally speaking we are interested in analyzing the limit:

$$\lim_{k \rightarrow +\infty} \mathbb{E} \|\pi_k - \tilde{\pi}_k\|_{TV}, \quad (3.13)$$

where  $\|\cdot\|_{TV}$  denotes the total variation norm. For given probabilistic measures  $\mu$ , and  $\nu$  the total variation norm  $\|\mu - \nu\|_{TV}$  is defined by:

$$\|\mu - \nu\|_{TV} := \sup_A (\mu - \nu)(A) + \sup_A (\nu - \mu)(A), \quad (3.14)$$

where the supremum is taken over all the sets  $A$  in the  $\sigma$ -field on which the probabilistic measures  $\mu, \nu$  are defined.

In case of nonlinear filtering the stability analysis requires an advanced measure theoretic approach that is out of scope of this chapter. A comprehensive overview of the stability properties and asymptotic analysis of nonlinear filtering methods is given in [Atar, 2011; Budhiraja, 2011; Chigansky et al., 2011; Crisan and Rozovskiĭ, 2011; Gland et al., 2011; Kleptsyna and Veretennikov, 2011; van Handel, 2010].

### 3.2.3 Optimal Filter for Linear Systems

A classical approach to the linear filtering problem was formulated in the innovative paper of Kalman [Kalman, 1960]. In recognition of that work, the algorithm presented therein has been called the *Kalman Filter* (KF) ever since.

#### Kalman Filter

The KF has been designed for systems with linear dynamics and additive zero-mean Gaussian noises. That is, the system equations (3.1)-(3.2) take the form

$$\mathbf{x}_{k+1} = \mathbf{F}_k \mathbf{x}_k + \mathbf{v}_k, \quad (3.15a)$$

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{w}_k, \quad (3.15b)$$

where  $\mathbf{F}_k$  and  $\mathbf{H}_k$  are matrices of appropriate dimensions, and  $\mathbf{v}_k$  and  $\mathbf{w}_k$  are uncorrelated zero-mean Gaussian variables with covariance matrices  $\mathbf{Q}_k$  and  $\mathbf{R}_k$ , respectively. This is the modern framework of the KF [Haykin, 1991; Ristic et al., 2004a], although it should be mentioned that the original formulation of Kalman [Kalman, 1960] was slightly different, in particular the measurement noise  $\mathbf{w}_k$  was not included. From a mathematical perspective the aforementioned assumptions lead to a restricted class of systems. However, such systems, used as approximations, are commonly found in engineering applications which makes the KF a very popular tool among practitioners ([Ljung and Gunnarsson, 1990; Matthies et al., 1989; Mehra, 1970; Welch and Bishop, 1995] and Chapter 18 of [Mendel, 1995a]).

For the stochastic process defined by (3.15), under the assumption that the initial density  $p_0$  is also Gaussian and uncorrelated with variables  $\mathbf{v}_k$  and  $\mathbf{w}_k$  [Ho and Lee, 1964], at each time step  $k$  the posterior density  $p(\mathbf{x}_k | \mathcal{Y}_k)$  is Gaussian. The KF recursively derives the predicted state density  $p(\mathbf{x}_k | \mathcal{Y}_{k-1})$  and the posterior density  $p(\mathbf{x}_k | \mathcal{Y}_k)$  as

$$p(\mathbf{x}_k | \mathcal{Y}_{k-1}) = \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1}), \quad (3.16a)$$

$$p(\mathbf{x}_k | \mathcal{Y}_k) = \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k}), \quad (3.16b)$$

where  $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$  is the density of a normally distributed random vector, with the mean  $\boldsymbol{\mu}$  and the covariance matrix  $\boldsymbol{\Sigma}$ , evaluated at  $\mathbf{x}$ . The estimated means and covariances in (3.16) follow the recursive relationship:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_{k-1} \hat{\mathbf{x}}_{k-1|k-1}, \quad (3.17)$$

$$\mathbf{P}_{k|k-1} = \mathbf{Q}_{k-1} + \mathbf{F}_{k-1} \mathbf{P}_{k-1|k-1} \mathbf{F}_{k-1}^T, \quad (3.18)$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{y}_k - \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1}), \quad (3.19)$$

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1}, \quad (3.20)$$

where  $\hat{\mathbf{x}}_{k-1|k-1}$  and  $\mathbf{P}_{k-1|k-1}$  are the mean and the covariance, respectively, of the Gaussian posterior  $p(\mathbf{x}_{k-1} | \mathcal{Y}_{k-1})$ ,  $\mathbf{I}$  denotes the identity matrix, and  $\mathbf{K}_k$  is a matrix given by:

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k)^{-1}. \quad (3.21)$$

### 3. NONLINEAR BAYESIAN FILTERING

---

In order to implement the KF one needs to specify the covariances  $\mathbf{Q}_k$  and  $\mathbf{R}_k$ . Often, these are unknown and have to be determined from data or by using prior knowledge. In [Mehra, 1970] methods for obtaining unbiased, consistent and asymptotically normal estimates of  $\mathbf{Q}_k$  and  $\mathbf{R}_k$  have been developed. These algorithms have been further improved in [Akesson et al., 2008; Alspach, 1972; Rajamani and Rawlings, 2009; Sangsuk-Iam and Bullock, 1990]. The effect of prefiltering upon the estimates of covariances has been studied in [Reynolds, 1990]. It has been noted that the precise knowledge of  $\mathbf{Q}_k$  is more critical than knowledge of  $\mathbf{R}_k$ , which often can be derived from sensor specifications. This is because incorrect values of the system noise covariance  $\mathbf{Q}_k$  can cause the filter to diverge [Sangsuk-Iam and Bullock, 1990; Welch and Bishop, 1995] and Chapter 18 of [Mendel, 1995a].

The KF is an unbiased estimator [Anderson and Moore, 1979]. Moreover, for the system defined by (3.15) the KF is an optimal filter, i.e., no algorithm can outperform (in the MMSE sense) the KF in this setting [Anderson and Moore, 1979]. Indeed, it can be shown [Anderson and Moore, 1979; Tichavsky et al., 1998] that the covariance matrix  $\mathbf{P}_{k|k}$  defined by (3.20) is equal to the inverse of the information matrix  $\mathbf{J}_k$  computed by (3.10), i.e.,

$$\mathbb{E}_{\mathbf{y}_{1:k}, \mathbf{x}_{1:k}} \left( (\hat{\mathbf{x}}_{k|k} - \mathbf{x}_k) (\hat{\mathbf{x}}_{k|k} - \mathbf{x}_k)^T \right) = \mathbf{P}_{k|k} = (\mathbf{J}_k)^{-1}, \quad (3.22)$$

hence, by (3.9), the estimate obtained by (3.19) provides the minimal estimation error. Alternatively, we can say that the KF provides the minimal variance estimator over all unbiased estimators designed for the system (3.15). However, the optimality of the KF only holds for linear systems with additive Gaussian noises whereas in many engineering applications, one has to work with nonlinear dynamics and possibly non-Gaussian noises. In such cases the KF becomes a suboptimal method that is outperformed by more advanced algorithms. It is also possible to derive the KF for systems under less restrictive conditions than those presented in this section, e.g., for systems influenced by nonzero mean variables  $\mathbf{v}_k$  and  $\mathbf{w}_k$ , or systems with correlated system and measurement noises. These are discussed in, e.g., Chapter 22 of [Mendel, 1995a].

#### Example: Random Walk Model

To illustrate the capabilities of the KF we apply it to a simple, yet useful ([Mulquiney et al., 1995; Sargan and Bhargava, 1983; Welch and Bishop, 1995], Chapter 3 of [Anderson and Moore, 1979], Chapter 4 of [Kushner and Dupuis, 2001]), zero-drift one-dimensional Gaussian random walk model with noisy measurements (a continuous-time analogue of this process is discussed in Chapter 6 of [Øksendal, 2003a]). The system equations are given by

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{v}_k, \quad (3.23a)$$

$$\mathbf{y}_k = \mathbf{x}_k + \mathbf{w}_k, \quad (3.23b)$$

with  $\mathbf{v}_k$  and  $\mathbf{w}_k$  being mutually independent zero-mean Gaussian variables with standard deviations  $\sqrt{\mathbf{Q}_k} = 0.01$  and  $\sqrt{\mathbf{R}_k} = 0.05$ , respectively. We start the simulations at the initial state  $\mathbf{x}_0 = 0.26$ , which was randomly chosen from the interval  $[-1, 1]$ , from which point the process evolves according to (3.23a) for 30 time steps. For each  $\mathbf{x}_k$  the corresponding measurement  $\mathbf{y}_k$  is generated according to (3.23b).

Clearly, the system (3.23) is linear, hence the KF, produces the optimal estimate  $\hat{\mathbf{x}}_{k|k}$ . To better illustrate the tracking abilities of the KF the filter is initialized at  $\hat{\mathbf{x}}_0 = 0$ , i.e., in the center of the interval  $[-1, 1]$  where the true state  $\mathbf{x}_0$  is expected to be, with the initial standard deviation  $\sqrt{\mathbf{P}_0} = 1$ . The results of the simulation are presented in Figure 3.1a.

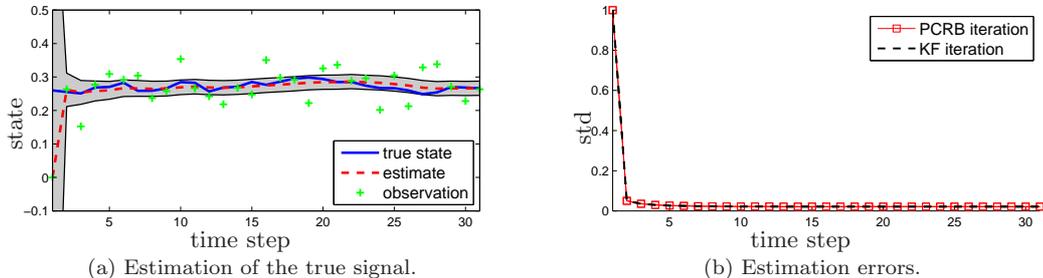


Figure 3.1: Kalman Filter applied to a random walk model. On the left: estimates (dashed line) of the true state (thick solid line) given the noisy measurements (+). The shaded area constitutes the one-sigma uncertainty interval  $[\hat{\mathbf{x}}_{k|k} - \sqrt{\mathbf{P}_{k|k}}, \hat{\mathbf{x}}_{k|k} + \sqrt{\mathbf{P}_{k|k}}]$ . On the right: standard deviation of the estimator obtained with the recursion (3.10) (solid line with square marks) and with the KF recursion (3.17)–(3.21) (dashed thick line).

We see that even though the filter starts from the initial position  $\hat{\mathbf{x}}_0 = 0$  different from the true initial state  $\mathbf{x}_0 = 0.26$  it quickly tracks the signal and, despite noisy measurements, closely follows it for the remainder of the simulation. Also, the uncertainty of the estimate quickly decreases from the initial value  $\sqrt{\mathbf{P}_0} = 1$  to the final  $\sqrt{\mathbf{P}_{30|30}} = 0.0213$ .

Now, we would like to confirm experimentally that the KF is indeed an optimal filter in a linear setting. If that is really the case, in light of (3.22), the variance  $\mathbf{P}_{k|k}$  computed from (3.17)–(3.21) should be approximately equal to the inverted information matrix  $(\mathbf{J}_k)^{-1}$  obtained by the recursion (3.10). For a linear system with additive noises the matrices defined in (3.11) take the form [Tichavsky et al., 1998]:

$$\mathbf{D}_k^{11} = \mathbf{F}_k^T (\mathbf{Q}_k)^{-1} \mathbf{F}_k, \quad (3.24a)$$

$$\mathbf{D}_k^{12} = \mathbf{F}_k^T (\mathbf{Q}_k)^{-1}, \quad (3.24b)$$

$$\mathbf{D}_k^{22} = (\mathbf{Q}_k)^{-1} + \mathbf{H}_{k+1}^T (\mathbf{R}_{k+1})^{-1} \mathbf{H}_{k+1}, \quad (3.24c)$$

with the initial condition given by

$$\mathbf{J}_0 = (\mathbf{P}_0)^{-1}. \quad (3.25)$$

In our example these matrices are reduced to scalar values

$$\mathbf{D}_k^{11} = 10^4, \quad (3.26a)$$

$$\mathbf{D}_k^{12} = 10^4, \quad (3.26b)$$

$$\mathbf{D}_k^{22} = 10^4 + 4 \cdot 10^2, \quad (3.26c)$$

with

$$\mathbf{J}_0 \approx 1. \quad (3.27)$$

Running the recursion (3.10) using the values (3.26) it is easy to check that for the random walk model the inverted information matrices  $(\mathbf{J}_k)^{-1}$  closely match the variance  $\mathbf{P}_{k|k}$  (see also Figure 3.1b).

### 3.3 Parametric Nonlinear Bayesian Filtering

In this section we review a number of approximation methods that belong to the class of Parametric Nonlinear Filters, i.e., methods for which the estimation problem has a solution in a finite dimensional parameter space. Within this class we distinguish three types of algorithms: Analytical Approximations, Statistical Approximations and Gaussian Sum Approximations.

To help the reader better understand the properties of the filters discussed we analyze their performance in several numerical experiments. For these experiments we use popular systems that have been extensively studied in the literature.

#### 3.3.1 Analytical Approximations

Within the framework of Section 3.2.3 (linear dynamics, additive Gaussian noises), the KF is the optimal filter. Unfortunately, when nonlinear or non-Gaussian problems are considered, no optimal solution (in the MMSE sense) exists, i.e., can be computed in closed analytical form (see [Ristic et al., 2004a] and references therein). However, throughout the years many suboptimal techniques [Kushner and Dupuis, 2001; Pugachev and Sinitsyn, 2001] have been developed for various classes of nonlinear stochastic dynamical systems. From this section onwards we focus on such estimation methods.

In this section we describe the historically firstly proposed *Extended Kalman Filter* (EKF) and its modification, the *Iterated Extended Kalman Filter* (IEKF). Both filters are analytical methods because the approximations of (3.4)–(3.5) are derived using the Taylor series expansion, a method that exploits the analytical structure of the functions  $\mathbf{f}_k$  and  $\mathbf{h}_k$ .

#### Extended Kalman Filter

The EKF [Mendel, 1995b; Ristic et al., 2004b; Welch and Bishop, 1995] is one of the most popular modifications of the KF and is designed to estimate the states of a nonlinear system. The main idea of the EKF algorithm is that at each time step the nonlinear state (3.1) and observation (3.2) models can be analytically approximated in order to obtain a linear system.

For sufficiently smooth functions  $\mathbf{f}_k$  and  $\mathbf{h}_k$  given the previous state estimate  $\hat{\mathbf{x}}_{k-1|k-1}$  and covariance  $\mathbf{P}_{k-1|k-1}$  the EKF approximates the right-hand side of (3.1)–(3.2) with the first-order Taylor series expansion around the points  $(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{0})$  and  $(\mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{0}), \mathbf{0})$ , respectively [Ristic et al., 2004b; Welch and Bishop, 1995]:

$$\mathbf{x}_k \approx \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{0}) + \mathbf{F}_{k-1} \cdot (\Delta \mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1|k-1}) + \mathbf{V}_{k-1} (\Delta \mathbf{v}_{k-1}) \quad (3.28a)$$

$$\mathbf{y}_k \approx \mathbf{h}_k(\mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{0}), \mathbf{0}) + \mathbf{H}_k \cdot (\Delta \mathbf{x}_k - \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{0})) + \mathbf{W}_k (\Delta \mathbf{w}_k) \quad (3.28b)$$

for every  $\Delta \mathbf{x}_{k-1}, \Delta \mathbf{x}_k, \Delta \mathbf{v}_{k-1}, \Delta \mathbf{w}_k$ , where

1.  $\mathbf{F}_{k-1}$  is the Jacobian matrix of the partial derivatives of  $\mathbf{f}_{k-1}$  with respect to the state variable  $\mathbf{x}$ , evaluated in  $(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{0})$ :

$$\mathbf{F}_{k-1} = \frac{\partial \mathbf{f}_{k-1}}{\partial \mathbf{x}}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{0}), \quad (3.29)$$

2.  $\mathbf{V}_{k-1}$  is the Jacobian matrix of the partial derivatives of  $\mathbf{f}_{k-1}$  with respect to the noise variable  $\mathbf{v}$ , evaluated in  $(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{0})$ :

$$\mathbf{V}_{k-1} = \frac{\partial \mathbf{f}_{k-1}}{\partial \mathbf{v}}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{0}), \quad (3.30)$$

### 3.3. Parametric Nonlinear Bayesian Filtering

---

3.  $\mathbf{H}_k$  is the Jacobian matrix of the partial derivatives of  $\mathbf{h}_k$  with respect to the state variable  $\mathbf{x}$ , evaluated in  $(\mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{0}), \mathbf{0})$ :

$$\mathbf{H}_k = \frac{\partial \mathbf{h}_k}{\partial \mathbf{x}}(\mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{0}), \mathbf{0}), \quad (3.31)$$

4.  $\mathbf{W}_k$  is the Jacobian matrix of the partial derivatives of  $\mathbf{h}_k$  with respect to the noise variable  $\mathbf{w}$ , evaluated in  $(\mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{0}), \mathbf{0})$ :

$$\mathbf{W}_k = \frac{\partial \mathbf{h}_k}{\partial \mathbf{w}}(\mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{0}), \mathbf{0}). \quad (3.32)$$

It can be easily seen that the right-hand side of both (3.28a) and (3.28b) are Gaussian random variables. Therefore, similarly to the KF, the predicted state and the posterior state densities take the form (3.16a) and (3.16b), respectively, with the means and the covariances as given in Algorithm 3.1.

---

#### Algorithm 3.1 Extended Kalman Filter

---

**Input:**  $\mathbf{P}_{k-1|k-1}$ ,  $\hat{\mathbf{x}}_{k-1|k-1}$ ,  $\mathbf{Q}_{k-1}$ , and  $\mathbf{R}_k$

**Prediction step:**

Compute matrices  $\mathbf{F}_{k-1}$ , and  $\mathbf{V}_{k-1}$  according to (3.29)–(3.30)

Compute the predicted mean  $\hat{\mathbf{x}}_{k|k-1}$ :

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{0})$$

Compute the predicted covariance  $\mathbf{P}_{k|k-1}$ :

$$\mathbf{P}_{k|k-1} = \mathbf{F}_{k-1} \mathbf{P}_{k-1|k-1} \mathbf{F}_{k-1}^T + \mathbf{V}_{k-1} \mathbf{Q}_{k-1} \mathbf{V}_{k-1}^T$$

**Update step:**

Compute matrices  $\mathbf{H}_k$  and  $\mathbf{W}_k$  according to (3.31)–(3.32)

Compute the Kalman gain  $\mathbf{K}_k$ :

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{W}_k \mathbf{R}_k \mathbf{W}_k^T)^{-1}$$

Compute the estimated mean  $\hat{\mathbf{x}}_{k|k}$ :

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{y}_k - \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}, \mathbf{0}))$$

Compute the estimated covariance  $\mathbf{P}_{k|k}$ :

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1}$$


---

The approximation (3.28) is accurate only if the following three assumptions hold:

- I. the noises  $\mathbf{v}_k$  and  $\mathbf{w}_k$  are lightly tailed, i.e., the norms of the covariance matrices  $\mathbf{V}_k$  and  $\mathbf{W}_k$  are small,
- II. the estimate  $\hat{\mathbf{x}}_{k-1|k-1}$  is approximately equal to the actual state of the system at time step  $k - 1$ ,
- III. the functions  $\mathbf{f}_k$  and  $\mathbf{h}_k$  do not exhibit severe nonlinear behavior.

The first two postulates, together with the fact that  $\mathbb{E}[\mathbf{v}_k] = \mathbf{0}$  and  $\mathbb{E}[\mathbf{w}_k] = \mathbf{0}$  justify the Taylor expansions around the aforementioned points, whereas the third one allows one to truncate the infinite series after the first derivative term.

Note that as far as real systems are concerned Postulate I seems reasonable. Indeed, in most applications the process and the measurement noises are bounded within narrow intervals (see Part II of [Ristic et al., 2004a], or Chapter 6 of [Tanizaki, 1996]). Postulate II simply states

### 3. NONLINEAR BAYESIAN FILTERING

---

that the estimator is accurate, meaning unbiased, and precise, meaning with small covariance matrix.

Postulate III is more critical. To understand why, recall that the approximation (3.28a) models the predicted state as a Gaussian random variable, whereas in reality a variable after the nonlinear transformation  $\mathbf{f}_k$  is no longer normally distributed. In case of mild nonlinearities (different measures of nonlinearity are reported in [Verlaan and Heemink, 2001] and [Ali-Löytty, 2008a]), the transformed variable can be accurately approximated by a Gaussian distribution. However, for a system that exhibits a strong nonlinear behavior the approximation is no longer feasible and might result in an inconsistent estimator. The influence of the linearization errors on the final EKF performance has been extensively studied in the literature [Ristic et al., 2004a; Tanizaki and Mariano, 1996; Xiong et al., 2008]. An informative case study is presented in Chapter 2 of [Ristic et al., 2004a], where the discussed nonlinear function is a transformation that converts a Gaussian random variable from polar to Cartesian coordinates. A similar example can be found in [Julier and Uhlmann, 1996], where the model of a vehicle moving along a circular arc is investigated. Also in this case the application of EKF leads to significant estimation errors.

Finally, similarly to the KF case, the EKF requires the covariance matrices  $\mathbf{Q}_k$  and  $\mathbf{R}_k$ . They can be derived from stochastic properties of the noises  $\mathbf{v}_k$  and  $\mathbf{w}_k$  or, if these are unknown, tuned from data [Bolognani et al., 2003; Chang and Tabaczynski, 1984].

#### Iterated Extended Kalman Filter

In order to improve the EKF the *Iterated Extended Kalman Filter* (IEKF) has been developed [Jazwinski, 1970; Tanizaki, 1996]. This algorithm has a strong resemblance to the conventional EKF. In fact, for both filters the linearization of the prediction function  $\mathbf{F}_k$  is derived in the same manner, and they differ only in the way in which the updated estimate is computed. The IEKF assumes that the measurement model is such that for every time step  $k$  the noise variable  $\mathbf{w}_k$  can be explicitly expressed as a function of  $\mathbf{y}_k$  and  $\mathbf{x}_k$ , i.e., for each  $k$  there exists a function  $\mathbf{g}_k$  such that:

$$\mathbf{w}_k = \mathbf{g}_k(\mathbf{y}_k, \mathbf{x}_k). \quad (3.33)$$

If the observation model has additive linear noises, i.e.,

$$\mathbf{h}_k(\mathbf{x}_k, \mathbf{w}_k) = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{H}_k \mathbf{w}_k, \quad (3.34)$$

with an invertible matrix  $\mathbf{H}_k$ , then

$$\mathbf{g}_k(\mathbf{y}_k, \mathbf{x}_k) = (\mathbf{H}_k)^{-1} (\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k)), \quad (3.35)$$

which is the scaled difference between the measured and the predicted variables. The IEKF linearizes  $\mathbf{g}_k$  around the updated state estimate  $\hat{\mathbf{x}}_{k|k}$  rather than around the predicted state estimate  $\hat{\mathbf{x}}_{k|k-1}$  as the EKF does. This is achieved by the following iteration (hence the name): the algorithm starts with a linearized model around the predicted estimate  $\hat{\mathbf{x}}_{k|k-1}$ , and uses it to compute the updated state estimate  $\hat{\mathbf{x}}_{k|k}^1$ . Then, the function  $\mathbf{g}_k$  is linearized around this newly obtained vector, and the new updated state estimate  $\hat{\mathbf{x}}_{k|k}^2$  is derived. This procedure is repeated until the iteration step  $i_0$  is reached such that  $\|\hat{\mathbf{x}}_{k|k}^{i_0} - \hat{\mathbf{x}}_{k|k}^{i_0-1}\| < \epsilon$ , where  $\epsilon$  is a predefined small number. This iteration, which is equivalent to the Gauss-Newton method [Bell and Cathey, 1993], is presented in Algorithm 3.2. The update algorithm has been derived, e.g., in Section 3.4 of [Tanizaki, 1996].

---

**Algorithm 3.2** IEKF: Update Iteration

---

**Input:**  $\epsilon, \mathbf{P}_{k|k-1}, \hat{\mathbf{x}}_{k|k-1}, \mathbf{R}_k, \mathbf{y}_k$   
Set the initial estimate:  $\hat{\mathbf{x}}_{k|k}^0 = \hat{\mathbf{x}}_{k|k-1}$   
Set the initial counter:  $i = 0$   
**repeat**  
  Augment the counter:  $i = i + 1$   
  Linearize the error model :  $\mathbf{H}_k^i = \frac{\partial \mathbf{g}_k}{\partial \mathbf{x}} \left( \mathbf{y}_k, \hat{\mathbf{x}}_{k|k}^{i-1} \right)$   
  Compute the Kalman gain:  
   $\mathbf{K}_k^i = \mathbf{P}_{k|k-1} \left( \mathbf{H}_k^i \right)^T \left( \mathbf{H}_k^i \mathbf{P}_{k|k-1} \left( \mathbf{H}_k^i \right)^T + \mathbf{R}_k \right)^{-1}$   
  Update the estimate:  
   $\hat{\mathbf{x}}_{k|k}^i = \hat{\mathbf{x}}_{k|k-1} - \mathbf{K}_k^i \left( \mathbf{g}_k \left( \mathbf{y}_k, \hat{\mathbf{x}}_{k|k}^{i-1} \right) + \mathbf{H}_k^i \left( \hat{\mathbf{x}}_{k|k-1} - \hat{\mathbf{x}}_{k|k}^{i-1} \right) \right)$   
**until**  $\| \hat{\mathbf{x}}_{k|k}^i - \hat{\mathbf{x}}_{k|k}^{i-1} \| < \epsilon$   
Set:  $i_0 = i$   
Set the updated estimate:  $\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k}^{i_0}$   
Set the covariance of the updated estimate:  
 $\mathbf{P}_{k|k} = \left( \mathbf{I} - \mathbf{K}_k^{i_0} \mathbf{H}_k^{i_0} \right) \mathbf{P}_{k|k-1}$

---

Note that in the case of a linear observation model with additive noises the IEKF is reduced to the standard EKF. The disadvantage of IEKF is that, due to the internal loop, it is numerically more involved than the EKF. Also, it has been argued that both IEKF and EKF perform similarly if the state is only partially observable [Lefebvre et al., 2004]. Informative examples of applications and comparison of the performance of the two filters are discussed in [Spingarn, 1987] and [Lefebvre et al., 2004].

#### Other EKF-like Algorithms

The accuracy of the EKF can be further improved by the addition of higher-order terms in the approximation (3.28). Better accuracy comes with the price of increased computational burden. Furthermore, although higher-order filters reduce the bias of the estimators [Wishner et al., 1969], in general, they cannot produce unbiased estimates [Tanizaki and Mariano, 1996].

Other variations of EKF that avoid gradient computations have been developed recently [Gosh et al., 2007; Saha and Roy, 2009]. Regarding these filters, two approaches can be distinguished: implicit methods, and explicit methods. In the implicit approach the problem of calculating a Jacobian is replaced by the one of finding a solution of an analytical equation (see [Gosh et al., 2007] and the references therein). In the explicit approach the nonlinear operator is linearized by means of Euler or Newmark expansion [Saha and Roy, 2009]. As presented in [Gosh et al., 2007] and [Saha and Roy, 2009], in certain situations, e.g., in case of jumps in parameter values, these filters achieve better performance than the conventional EKF.

#### Example

To illustrate the difference in the performance between EKF and IEKF, let us investigate a simple two-dimensional nonlinear system defined by

$$\mathbf{x}_{k+1}(1) = 0.1 (\mathbf{x}_k(1))^2 - 2\mathbf{x}_k(1) + 20 + \mathbf{v}_k(1), \quad (3.36a)$$

$$\mathbf{x}_{k+1}(2) = \mathbf{x}_k(1) + 0.3\mathbf{x}_k(2) - 3 + \mathbf{v}_k(2), \quad (3.36b)$$

### 3. NONLINEAR BAYESIAN FILTERING

and

$$\mathbf{y}_k(1) = (\mathbf{x}_k(1))^2 + (\mathbf{x}_k(2))^2 + \mathbf{w}_k(1), \quad (3.37a)$$

$$\mathbf{y}_k(2) = 3(\mathbf{x}_k(2))^2/\mathbf{x}_k(1) + \mathbf{w}_k(2) \quad (3.37b)$$

Equations (3.36)–(3.37) constitute a system that is a modification of the case studied in [Lefebvre et al., 2004]. The system is nonlinear in both the state model (the second order term in (3.36a)) and the observation model. Furthermore, both the state and the observation models are influenced by mutually independent additive Gaussian noises  $\mathbf{v}_k$  and  $\mathbf{w}_k$  with covariance matrices  $\mathbf{Q} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$  and  $\mathbf{R} = \begin{bmatrix} 1 & 0 \\ 0 & 10 \end{bmatrix}$ , respectively.

For the purpose of comparison, starting from the initial state  $\mathbf{x}_0 = [10 \ 10]^T$ , we have generated the trajectory  $\mathbf{x}_{1:20} = (\mathbf{x}_1, \dots, \mathbf{x}_{20})$  with the corresponding observations  $\mathcal{Y}_{20}$ , according to (3.36) and (3.37) respectively. Figure 3.2 compares the estimates obtained by EKF and IEKF aiming to reproduce the trajectory  $\mathbf{x}_{1:20}$  from the generated measurements  $\mathcal{Y}_{20}$ . Both filters are initialized from the actual state of the system, i.e., from  $\mathbf{x}_0 = [10 \ 10]^T$ , each having the same initial uncertainty about the true state  $\mathbf{P}_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ . Furthermore, the parameter  $\epsilon$ , which is used in Algorithm 3.2, is set to  $\epsilon = 0.0001$ .

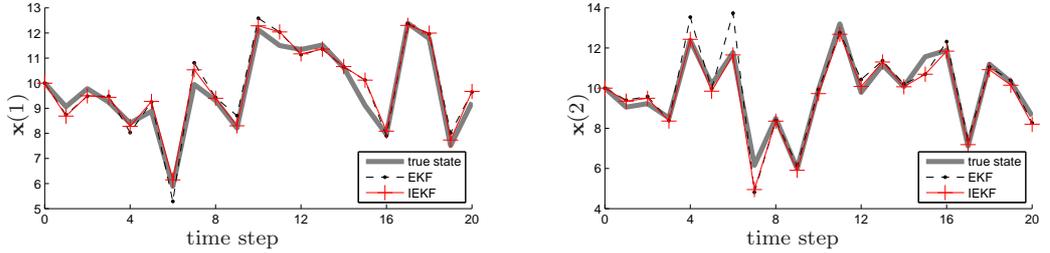


Figure 3.2: The EKF estimate (thin solid line) and the IEKF estimate (dashed line) vs the sample trajectory  $\mathbf{x}_{0:20}$  generated from the system (3.36)–(3.37) (thick solid line).

From Figure 3.2 it can be observed that most of the time the two nonlinear filters behave similarly. However, in some cases the IEKF tracks the actual state of the system more closely than the EKF does.

Let us now compute the PCRb for the system (3.36)–(3.37). It can be shown [Song et al., 2011; Tichavsky et al., 1998] that in case of additive Gaussian noises the matrices (3.11) are given by:

$$\mathbf{D}_k^{11} = \mathbb{E}_{\mathbf{x}_{1:k}} \left( (\nabla_{\mathbf{x}_k} \mathbf{f}_k^T(\mathbf{x}_k)) (\mathbf{Q}_k)^{-1} (\nabla_{\mathbf{x}_k} \mathbf{f}_k^T(\mathbf{x}_k))^T \right), \quad (3.38a)$$

$$\mathbf{D}_k^{12} = -\mathbb{E}_{\mathbf{x}_{1:k}} \left( (\nabla_{\mathbf{x}_k} \mathbf{f}_k^T(\mathbf{x}_k)) (\mathbf{Q}_k)^{-1} \right), \quad (3.38b)$$

$$\mathbf{D}_k^{22} = (\mathbf{Q}_k)^{-1} + \mathbb{E}_{\mathbf{x}_{1:k+1}} \left( (\nabla_{\mathbf{x}_{k+1}} \mathbf{h}_{k+1}^T(\mathbf{x}_{k+1})) (\mathbf{R}_{k+1})^{-1} (\nabla_{\mathbf{x}_{k+1}} \mathbf{h}_{k+1}^T(\mathbf{x}_{k+1}))^T \right). \quad (3.38c)$$

The derivatives in (3.38) are evaluated in the true states of the system and the expectations are obtained by Monte Carlo simulation [Ripley, 1987] averaged over 10000 realizations of the independent trajectories of the system, with the initial distribution  $p_0 = \mathcal{N}(\mathbf{x}_0, \mathbf{P}_0)$ . The initial

information matrix is given by

$$\mathbf{J}_0 = (\mathbf{P}_0)^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \quad (3.39)$$

Figure 3.3 shows the square roots of the theoretical PCRB for states  $x_1$  and  $x_2$ . Furthermore, the theoretical lower bounds are compared with the *Root Mean Squared Errors* (RMSE) obtained from 10000 Monte Carlo runs of the system (3.36)–(3.37) with the same initial distribution and the same noise levels. From Figure 3.3 it can be observed that the IEKF has a lower RMSE than the EKF.

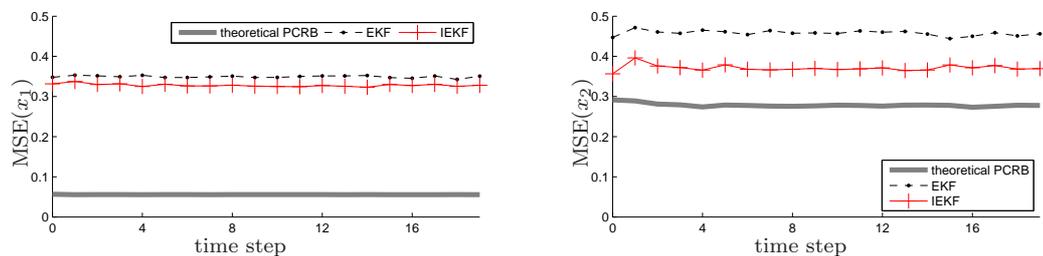


Figure 3.3: RMSE of the estimators given by EKF (dashed line) and IEKF (thin solid line) compared with the squared roots of the theoretical PCRB (thick solid line) for the states  $x_1$  (left) and  $x_2$  (right).

### 3.3.2 Statistical Approximations

In this section we discuss an alternative approach to the nonlinear approximation problem, namely the statistical approach. Contrary to the methods presented in Section 3.3.1, the filters described in the current section do not use the Taylor series expansion. Instead, we are interested in statistical information that can be extracted from the system (3.1)–(3.3) and used afterward to estimate (3.4)–(3.5). All the filters discussed within this section can be considered as part of the general class of *linear regression Kalman Filters* (LRKFs).

The LRKF have been proposed by several authors [Ito and Xiong, 2000; Julier and Uhlmann, 1996; Julier et al., 2000; Lefebvre et al., 2004] and [Wan and van der Merwe, 2000]. Similarly to the EKF, these filters approximate the prediction and the posterior density as Gaussian densities, hence the formulas (3.16a) and (3.16b) still hold. However, the approximations of  $p(\mathbf{x}_k | \mathcal{Y}_{k-1})$  and  $p(\mathbf{x}_k | \mathcal{Y}_k)$  are obtained by means of statistical regression rather than through analytical approximations of the nonlinear functions  $\mathbf{f}_k$  and  $\mathbf{h}_k$  as in the EKF setting. The motivation for this approach can be intuitively expressed as follows: *With a fixed number of parameters it is easier to approximate a Gaussian distribution than it is to approximate an arbitrary nonlinear function* [Julier et al., 2000]. The general idea is to represent the a priori distributions by a set of deterministically chosen representative points and weights that completely capture the mean and the covariance of the Gaussian distribution and then use those points in the prediction and the update steps of the filter. This resembles the Monte Carlo approach. However, unlike Monte Carlo methods, the LRKFs are, in general, numerically less expensive since the samples are not drawn at random and the number of the required points is relatively small when compared to the number of samples that are generated by Monte Carlo algorithms. LRKFs achieve better accuracy than the EKF since the representative points

### 3. NONLINEAR BAYESIAN FILTERING

---

propagated through the nonlinear transformation capture the mean and covariance of the actual distribution up to the second order of nonlinearity [Julier et al., 2000].

There are many methods for the choice of the representative points and their weights, and the three most popular ones are discussed in this section. However, for the moment, let us not be concerned with any particular method of obtaining the representative points, but rather focus on the general framework of the LRKF.

The estimation proceeds as follows. At time step  $k-1$  the approximation of the posterior distribution is given by a Gaussian variable  $\mathcal{N}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1})$  and the noise  $\mathbf{v}_{k-1}$  is assumed to be distributed according to  $\mathcal{N}(\mathbf{0}, \mathbf{Q}_{k-1})$ . Both  $\mathbf{x}_{k-1}$  and  $\mathbf{v}_k$  are assumed to be uncorrelated and Gaussian. Therefore, the augmented state variable  $[\mathbf{x}_{k-1} \ \mathbf{v}_{k-1}]^T$  is also Gaussian with the mean  $\boldsymbol{\mu}_{k-1}^a$  and the covariance  $\mathbf{P}_{k-1}^a$  given by:

$$\boldsymbol{\mu}_{k-1}^a = \begin{bmatrix} \hat{\mathbf{x}}_{k-1|k-1} \\ \mathbf{0} \end{bmatrix}, \quad \mathbf{P}_{k-1}^a = \begin{bmatrix} \mathbf{P}_{k-1|k-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_{k-1} \end{bmatrix}.$$

The probability distribution  $\mathcal{N}(\boldsymbol{\mu}_{k-1}^a, \mathbf{P}_{k-1}^a)$  is encoded in the sequence  $\{(\mathbf{x}_{k-1}^i, \omega_{k-1}^i)\}_{i=1}^N$  that pairs each representative point  $\mathbf{x}_{k-1}^i$  with its weight  $\omega_{k-1}^i$ . The predicted state density  $p(\mathbf{x}_k | \mathcal{Y}_{k-1})$  is approximated by  $\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1})$ , where the mean and covariance are computed as follows:

$$\hat{\mathbf{x}}_{k|k-1} = \sum_{i=1}^N \omega_{k-1}^i \mathbf{f}_{k-1}(\mathbf{x}_{k-1}^i, \mathbf{0}), \quad (3.40)$$

$$\mathbf{P}_{k|k-1} = \sum_{i=1}^N \omega_{k-1}^i (\mathbf{f}_{k-1}(\mathbf{x}_{k-1}^i, \mathbf{0}) - \hat{\mathbf{x}}_{k|k-1}) (\mathbf{f}_{k-1}(\mathbf{x}_{k-1}^i, \mathbf{0}) - \hat{\mathbf{x}}_{k|k-1})^T \quad (3.41)$$

The distribution of the predicted measurement is obtained in a similar manner as the distribution of the predicted state. Namely, the noise  $\mathbf{w}_k$  is assumed to be zero-mean Gaussian with the covariance matrix  $\mathbf{R}_k$ , and independent of the state  $\mathbf{x}_k$ . Therefore, the variable  $[\mathbf{x}_k \ \mathbf{w}_k]^T$  is also Gaussian with the mean  $\boldsymbol{\mu}_k^a$  and the covariance  $\mathbf{P}_k^a$  given by:

$$\boldsymbol{\mu}_k^a = \begin{bmatrix} \hat{\mathbf{x}}_{k|k-1} \\ \mathbf{0} \end{bmatrix}, \quad \mathbf{P}_k^a = \begin{bmatrix} \mathbf{P}_{k|k-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_k \end{bmatrix}.$$

Next, the set of representative points and weights  $\{(\mathbf{x}_{k|k-1}^i, \omega_{k|k-1}^i)\}_{i=1}^N$  that approximate the distribution  $\mathcal{N}(\boldsymbol{\mu}_k^a, \mathbf{P}_k^a)$  is derived. The estimate of the measurement is then given by:

$$\hat{\mathbf{y}}_{k|k-1} = \sum_{i=1}^N \omega_{k|k-1}^i \mathbf{h}_k(\mathbf{x}_{k|k-1}^i, \mathbf{0}). \quad (3.42)$$

Finally the mean and covariance of the normal density that approximates the posterior  $p(\mathbf{x}_k | \mathcal{Y}_k)$  are computed as follows:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}), \quad (3.43)$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{P}_{xy} \mathbf{K}_k^T, \quad (3.44)$$

where the Kalman gain  $\mathbf{K}_k$  and the covariances  $\mathbf{P}_{xy}$  and  $\mathbf{P}_{yy}$  are computed as:

$$\mathbf{P}_{xy} = \sum_{i=1}^N \omega_{k|k-1}^i \left( \mathbf{x}_{k|k-1}^i - \hat{\mathbf{x}}_{k|k-1} \right) \left( \mathbf{h}_k(\mathbf{x}_{k|k-1}^i) - \hat{\mathbf{y}}_{k|k-1} \right)^T, \quad (3.45)$$

$$\mathbf{P}_{yy} = \sum_{i=1}^N \omega_{k|k-1}^i \left( \mathbf{h}_k(\mathbf{x}_{k|k-1}^i) - \hat{\mathbf{y}}_{k|k-1} \right) \left( \mathbf{h}_k(\mathbf{x}_{k|k-1}^i) - \hat{\mathbf{y}}_{k|k-1} \right)^T, \quad (3.46)$$

$$\mathbf{K}_k = \mathbf{P}_{xy} (\mathbf{P}_{yy})^{-1}. \quad (3.47)$$

The LRKF algorithm can be simplified for systems with additive noises, i.e., for systems where the functions  $\mathbf{f}_k$  and  $\mathbf{h}_k$  have the form:

$$\mathbf{f}_k(\mathbf{x}_k, \mathbf{v}_k) = \mathbf{f}_k(\mathbf{x}_k) + \mathbf{F}_k \mathbf{v}_k, \quad (3.48a)$$

$$\mathbf{h}_k(\mathbf{x}_k, \mathbf{w}_k) = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{H}_k \mathbf{w}_k, \quad (3.48b)$$

where both  $\mathbf{F}_k$  and  $\mathbf{H}_k$  are linear matrices. For such system one starts from computing the representative points and weights that approximate the distribution  $\mathcal{N}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1})$ . Next, the predicted state  $\hat{\mathbf{x}}_{k|k-1}$  is computed according to (3.40). In order to compute the covariance of the predicted state  $\mathbf{P}_{k|k-1}$  the right-hand side of (3.41) is modified by adding the term  $\mathbf{F}_{k-1} \mathbf{Q}_{k-1} \mathbf{F}_{k-1}^T$  which corresponds to the influence of the noise  $\mathbf{v}_{k-1}$  [Ristic et al., 2004a]:

$$\mathbf{P}_{k|k-1} = \sum_{i=1}^N \omega_{k-1}^i \left( \mathbf{f}_{k-1}(\mathbf{x}_{k-1}^i) - \hat{\mathbf{x}}_{k|k-1} \right) \left( \mathbf{f}_{k-1}(\mathbf{x}_{k-1}^i) - \hat{\mathbf{x}}_{k|k-1} \right)^T + \mathbf{F}_{k-1} \mathbf{Q}_{k-1} \mathbf{F}_{k-1}^T. \quad (3.49)$$

The next step is to approximate the distribution  $\mathcal{N}(\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1})$  by the set of representative points and weights. The procedure of obtaining the final estimates of  $\hat{\mathbf{x}}_{k|k}$  and  $\mathbf{P}_{k|k}$  is similar to the one described by equations (3.42)–(3.46). The only difference is that the transformed covariance of the observation noise  $\mathbf{w}_k$ , i.e.,  $\mathbf{H}_k \mathbf{R}_k \mathbf{H}_k^T$  has to be added to the right-hand side of (3.46). Therefore,  $\mathbf{P}_{yy}$  is given by [Ristic et al., 2004a]:

$$\mathbf{P}_{yy} = \sum_{i=1}^N \omega_{k-1}^i \left( \mathbf{h}_k(\mathbf{x}_{k|k-1}^i) - \hat{\mathbf{y}}_{k|k-1} \right) \left( \mathbf{h}_k(\mathbf{x}_{k|k-1}^i) - \hat{\mathbf{y}}_{k|k-1} \right)^T + \mathbf{H}_k \mathbf{R}_k \mathbf{H}_k^T \quad (3.50)$$

Note that the dimensions of the Gaussian variables  $\mathcal{N}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1})$  and  $\mathcal{N}(\hat{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1})$  are lower than the dimensions of the variables  $\mathcal{N}(\boldsymbol{\mu}_{k-1}^a, \mathbf{P}_{k-1}^a)$  and  $\mathcal{N}(\boldsymbol{\mu}_k^a, \mathbf{P}_k^a)$  approximated using the general algorithm. Therefore, a smaller number of representative points is required, and consequently fewer nonlinear transformations have to be performed. Instead, they are replaced by linear operations:  $\mathbf{F}_{k-1} \mathbf{Q}_{k-1} \mathbf{F}_{k-1}^T$  and  $\mathbf{H}_k \mathbf{R}_k \mathbf{H}_k^T$ .

It has been observed [Ito and Xiong, 2000] that the performance of the filter given by (3.40)–(3.46) strongly depends on the choice of the representative points. In the following we review the methods that have been proposed in the recent years. In order to keep the algorithms simple, we focus on filters designed for dynamical systems with additive noises. We motivate this choice by the fact that LRKF equations for systems with non-additive noises are conceptually identical. We start by describing the most popular LRKF, i.e., the *Unscented Kalman Filter* (UKF) and its variations. Next, other types of LRKF are discussed, namely the *Gauss-Hermite Filter* (GHF), and the *Central Difference Filter* (CDF). Finally, all the aforementioned filters are illustrated on an example.

### 3. NONLINEAR BAYESIAN FILTERING

---

#### Unscented Kalman Filter

Before we proceed to the detailed description of the UKF framework, we start with explaining the *Unscented Transformation* (UT) [Julier and Uhlmann, 1996]. This is a method of selecting representative points and weights that approximate a variable that undergoes a nonlinear transformation. The UKF uses the UT in a dynamic framework to obtain the approximations of the predicted state density and the predicted update density.

The UT is a general method for approximating the distribution of a Gaussian random variable that undergoes a nonlinear transformation. Let  $\mathbf{x}$  be such a variable, with mean  $\bar{\mathbf{x}}$  and covariance  $\mathbf{P}_{\mathbf{x}}$ , and let  $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^p$  be an arbitrary nonlinear function. The objective is to compute the statistics of the random variable  $\mathbf{y}$  defined as:

$$\mathbf{y} = \mathbf{g}(\mathbf{x}). \quad (3.51)$$

In order to do that, first one has to generate a set  $\Sigma = \{\sigma_i\}$  of *sigma points*, i.e., a set that is of zero sample mean and the points of this set have sample covariance equal to  $\mathbf{P}_{\mathbf{x}}$ . For the  $n$ -dimensional variable  $\mathbf{x}$ ,  $2n$  sigma points are computed as rows (or columns) of the matrix  $\pm\sqrt{(n+\lambda)\mathbf{P}_{\mathbf{x}}}$ , where  $\lambda = \alpha^2(n+\kappa) - n$  with a spread parameter  $\alpha$  and a scaling factor  $\kappa$ . The common choice for the spread parameter is  $\alpha = 1$  [Wan and Merwe, 2001] in which case  $\lambda = \kappa$  [Julier and Uhlmann, 1996; Julier et al., 2000].

The set  $\Sigma$  has the same mean and covariance as a zero mean Gaussian variable with a covariance matrix  $\mathbf{P}_{\mathbf{x}}$ . Furthermore, since it is symmetric, all the odd central moments are equal to zero as is the case with every zero mean Gaussian distribution. Therefore, the first three sample moments of  $\Sigma$  are equal to the theoretical moments of the variable  $\mathbf{x}$ . Hence, the approximation errors can occur only in the fourth and higher moments. The representative points of the distribution of the variable  $\mathbf{x}$  are generated by a translation of each sigma point by  $\bar{\mathbf{x}}$  and an assignment of appropriate weights [Julier and Uhlmann, 1996; Wan and Merwe, 2001]:

$$\begin{aligned} \mathbf{x}_0 &= \bar{\mathbf{x}} & \omega_0 &= \frac{\lambda}{n+\lambda}, \\ \mathbf{x}_i &= \bar{\mathbf{x}} + \sigma_i & \omega_i &= \frac{1-\omega_0}{2n}. \end{aligned}$$

The distribution of the transformed random variable  $\mathbf{y} = \mathbf{g}(\mathbf{x})$  is then represented by the set  $\{(\mathbf{g}(\mathbf{x}_i), \omega_i)\}_{i=0}^{2n}$ .

The errors in the calculation of the mean and covariance of  $\mathbf{y}$  are of fourth order in case of Gaussian inputs [Julier and Uhlmann, 1996] and of third order in case of non-Gaussian inputs [Wan and Merwe, 2001]. The approximation accuracy can be further improved by an appropriate choice of a scaling factor  $\kappa$  [Julier and Uhlmann, 1996]. The popular choice is  $\kappa = 3 - n$  [Julier et al., 2000; Wan and Merwe, 2001]. Setting  $\kappa = 0$  leads to the Cubature Kalman Filter introduced in [Arasaratnam and Haykin, 2009; Arasaratnam et al., 2010]. Further improvements in the quality of estimation can be achieved with the adaptive selection of  $\kappa$  which is done by the Adaptive UKF [Duník et al., 2010]. If available, the information on the higher order moments of the estimated variable can be used to modify the weight  $\omega_0$ , which reduces the higher-order errors of the UT [Julier and Uhlmann, 2004a,b; Wan and Merwe, 2001]. It is also possible to capture higher moments of the true distribution by augmenting the number of sigma points used in the approximation [Julier, 2002; Julier and Uhlmann, 2004a,b; Tenne and Singh, 2003]. For instance,  $2n^2 + 1$  sigma points are required to match the first four moments of a Gaussian distribution [Julier and Uhlmann, 2004a,b]. The accuracy of the approximation might further increase if the sigma points are scaled, so that all the sigma points lay in an

---

### 3.3. Parametric Nonlinear Bayesian Filtering

---

appropriate ellipsoid centered at the mean [Julier, 2002; Julier and Uhlmann, 2004a,b], or on the  $1\sigma$ ,  $2\sigma$  and  $3\sigma$  contours [Wu et al., 2004]. The  $k\sigma$  contour is the boundary of the ellipsoid defined by  $k\sqrt{\mathbf{P}_x}$ . The latter method requires  $6n + 1$  sigma points. The purpose of the scaling is to concentrate all the sigma points in the area of the highest probability.

Note that there are infinitely many square roots of the matrix  $\sqrt{\mathbf{P}_x}$  that can be chosen to compute the  $k\sigma$  contour [Julier and Uhlmann, 1996]. Therefore, the improvement of the computational properties of the UT is possible by the choice of an efficient numerical method for matrix square root computation. The most popular algorithm is the Cholesky decomposition, but other techniques, such as the more robust, but also more computationally involved, singular value decomposition can be used [Arasaratnam et al., 2007]. The computational efficiency of the UT can be further increased by reducing the number of sigma points that need to be generated in order to capture the desired properties of a distribution of the investigated random variable [Julier and Uhlmann, 2002, 2004a,b]. The minimal number of sigma points that is required to capture the mean and covariance is  $n + 1$ . The computational complexity of the UT grows linearly with the number of dimensions. However, with increasing dimensions of the state the accuracy of the UT approximation decreases [Honkela, 2004].

The UKF employs the UT at each filtering step following the procedure described in Algorithm 3.3.

---

#### Algorithm 3.3 Unscented Kalman Filter

---

**Input:**  $\kappa$ ,  $\mathbf{P}_{k-1|k-1}$ ,  $\hat{\mathbf{x}}_{k-1|k-1}$

Compute the sigma points  $\sigma_i$  as the columns of the matrix:  $\sqrt{(n + \kappa)\mathbf{P}_{k-1|k-1}}$

**Prediction step:**

Set the central point:  $\mathbf{x}_{k-1}^0 = \hat{\mathbf{x}}_{k-1|k-1}$

Set the central weight:  $\omega_{k-1}^0 = \frac{\kappa}{n + \kappa}$

**for**  $i = 1, \dots, 2n$  **do**

    Compute the representative points:

$\mathbf{x}_{k-1}^i = \hat{\mathbf{x}}_{k-1|k-1} \pm \sigma_i$

    Assign weights:  $\omega_{k-1}^i = \frac{1}{2(n + \kappa)}$

**end for**

Compute the predicted mean  $\hat{\mathbf{x}}_{k|k-1}$  using (3.40)

Compute the predicted covariance  $\mathbf{P}_{k|k-1}$  using (3.49)

**Update step:**

**for**  $i = 0, \dots, 2n$  **do**

    Compute the representative points:

$\mathbf{x}_{k|k-1}^i = \mathbf{f}_k(\mathbf{x}_{k-1}^i)$

    Assign weights:  $\omega_{k|k-1}^i = \omega_{k-1}^i$

**end for**

Compute the estimated measurement  $\hat{\mathbf{y}}_{k|k-1}$  using (3.42)

Compute the covariance of the predicted observation  $\mathbf{P}_{yy}$  using (3.50)

Compute the cross-covariance of the predicted observation and the predicted state  $\mathbf{P}_{xy}$  using (3.45)

Compute the Kalman gain  $\mathbf{K}_k$  using (3.47)

Compute the estimated mean  $\hat{\mathbf{x}}_{k|k}$  using (3.43)

Compute the estimated covariance  $\mathbf{P}_{k|k}$  using (3.44)

---

### 3. NONLINEAR BAYESIAN FILTERING

---

#### Gauss-Hermite Filter

An alternative method for determining the representative points with their weights is employed in the Gauss-Hermite Filter (GHF). The GHF is a Gaussian filter that utilizes the Gaussian-Hermite quadrature rule. This is an approximation technique used for evaluating an integral  $I$  of the form:

$$I = \int_{\mathbb{R}^n} f(\mathbf{x}) \frac{1}{(2\pi)^{n/2}} \exp\left(-\frac{\|\mathbf{x}\|^2}{2}\right) d\mathbf{x}, \quad (3.52)$$

where  $f$  is a given nonlinear function. In other words,  $I$  is the expectation of a standard normal variable propagated through the nonlinear function  $f$ . The integral above is approximated by the  $m$ -th order quadrature rule  $I_m$ :

$$\begin{aligned} I_m &= \sum_{i_1=1}^m \dots \sum_{i_n=1}^m \omega_{i_1} \dots \omega_{i_n} f(x_{i_1}, \dots, x_{i_n}) \\ &= \sum_{i=1}^{m^n} \omega_i f(\mathbf{x}_i), \end{aligned} \quad (3.53)$$

where for each  $1 \leq i \leq m^n$  the following holds:  $\mathbf{x}_i = (x_{i_1}, \dots, x_{i_n})^T$  and  $\omega_i = \prod_{j=1}^n \omega_{i_j}$ .

Suppose that  $\mathbf{J}$  is a symmetric tridiagonal matrix with zeros on the diagonal and the other entries defined by:

$$\mathbf{J}_{i,j} = \begin{cases} \sqrt{i/2}, & j = i + 1 \\ 0, & \text{otherwise} \end{cases}$$

For each  $1 \leq j \leq n$  the one dimensional  $m$ -th order quadrature rule  $\{(x_l, \omega_l)\}_{l=1}^m$  is computed in two steps [Golub and Welsch, 1969]. First, the quadrature point  $x_l$  is defined as the  $l$ -th eigenvalue  $\epsilon_l$  of the matrix  $\mathbf{J}$ , multiplied by  $\sqrt{2}$ . Next, the corresponding weight  $\omega_l$  is set to be equal to the square of the first element of the normalized  $l$ -th eigenvector  $\mathbf{v}_l$  of  $\mathbf{J}$ . To summarize:

$$x_l = \sqrt{2}\epsilon_l, \quad (3.54)$$

$$\omega_l = ((\mathbf{v}_l)_1)^2 \quad (3.55)$$

It is well known that the precision of the estimate increases with the order of the quadrature [Pomorski, 2006]. However, at the same time the computational burden grows with the rate  $m^n$ . Indeed, by (3.53), the computation of  $I_m$  requires  $m^n$  function evaluations, i.e.,  $m^n$  representative points need to be computed. Therefore, even for moderate state dimensions  $n$ , a higher-order GHF ( $m > 5$ ) means a significant computational load, which makes it impractical for online applications. Furthermore, for  $m > 1$  and large  $n$  the Gauss-Hermite quadrature rule is numerically more involved than the UT. In the special case of  $\kappa = 2$  and  $n = 1$  the UT matches  $I_3$ .

Note that this algorithm can easily be generalized for Gaussian variables with arbitrary mean  $\mu$  and covariance  $\Sigma$ , simply by replacing  $f$  with  $\tilde{f}(\mathbf{x}) = f(\sqrt{\Sigma}^T \mathbf{x} + \mu)$ .

The GHF utilizing the  $m$ -th order quadrature rule is presented in Algorithm 3.4.

#### Central Difference Filter

To choose the representative points the Central Difference Filter (CDF) or Divided Difference Filter [Lefebvre et al., 2004; Nørgaard et al., 2000] uses a different method than the previously

---

**Algorithm 3.4** Gauss-Hermite Filter
 

---

**Input:**  $\mathbf{P}_{k-1|k-1}$ ,  $\hat{\mathbf{x}}_{k-1|k-1}$ 

 Compute the one dimensional quadrature rule  $\{(x_i, \omega_i)\}_{i=1}^m$  using (3.54)–(3.55).

**for**  $1 \leq i_1, \dots, i_n \leq m$  **do**

 Compute the representative points  $\mathbf{x}_i = (x_{i_1}, \dots, x_{i_n})^T$ ,

 Compute the corresponding weights  $\omega_i = \prod_{j=1}^n \omega_{i_j}$ 
**end for**
**Prediction step:**

 Factorize the posterior covariance:  $\mathbf{P}_{k-1|k-1} = \mathbf{S}^T \mathbf{S}$ 
**for**  $i = 1, \dots, m$  **do**

Compute the representative points:

 $\mathbf{x}_{k-1}^i = \mathbf{S}^T \mathbf{x}_i + \hat{\mathbf{x}}_{k-1|k-1}$ 

 Assign weights:  $\omega_{k-1}^i = \omega_i$ 
**end for**

 Compute the predicted mean  $\hat{\mathbf{x}}_{k|k-1}$  according to (3.40)

 Compute the predicted covariance  $\mathbf{P}_{k|k-1}$  according to (3.41)

**Update step:**

 Factorize the predicted covariance:  $\mathbf{P}_{k|k-1} = \tilde{\mathbf{S}}^T \tilde{\mathbf{S}}$ 
**for**  $i = 1, \dots, m$  **do**

Compute the representative points:

 $\mathbf{x}_{k|k-1}^i = \tilde{\mathbf{S}}^T \mathbf{x}_i + \hat{\mathbf{x}}_{k|k-1}$ 

 Assign weights:  $\omega_{k|k-1}^i = \omega_i$ 
**end for**

 Compute the estimated measurement  $\hat{\mathbf{y}}_{k|k-1}$  using (3.42)

 Compute the covariance of the predicted observation  $\mathbf{P}_{yy}$  using (3.50)

 Compute the cross-covariance of the predicted observation and the predicted state  $\mathbf{P}_{xy}$  using (3.45)

 Compute the Kalman gain  $\mathbf{K}_k$  using (3.47)

 Compute the estimated mean  $\hat{\mathbf{x}}_{k|k}$  using (3.43)

 Compute the estimated covariance  $\mathbf{P}_{k|k}$  using (3.44)
 

---

discussed UKF and GHF. The CDF algorithm is based on the central difference approximation of the integral (3.52). The basic feature of this method is to approximate the nonlinear function  $f$  with a quadratic function  $P_2$  defined by:

$$P_2(\mathbf{x}) = f(\mathbf{0}) + \mathbf{a}\mathbf{x} + \frac{1}{2}\mathbf{x}^T \mathbf{H}\mathbf{x}, \quad (3.56)$$

where the vector  $\mathbf{a} = (a_i)$  and the symmetric matrix  $\mathbf{H} = (\mathbf{H}_{i,j})$  are given by [Ito and Xiong, 2000]:

$$a_i = \frac{f(\mathbf{h}\mathbf{e}_i) - f(-\mathbf{h}\mathbf{e}_i)}{2h}, \quad 1 \leq i \leq n \quad (3.57a)$$

$$\mathbf{H}_{i,i} = \frac{f(\mathbf{h}\mathbf{e}_i) - 2f(\mathbf{0}) + f(-\mathbf{h}\mathbf{e}_i)}{h^2}, \quad 1 \leq i \leq n \quad (3.57b)$$

$$\mathbf{H}_{i,j} = \frac{f(\mathbf{h}\mathbf{e}_i + \mathbf{h}\mathbf{e}_j) - f(-\mathbf{h}\mathbf{e}_i) - f(-\mathbf{h}\mathbf{e}_j) + f(\mathbf{0})}{h^2}, \quad 1 \leq i < j \leq n \quad (3.57c)$$

Here  $h > 0$  is a chosen step size and  $(\mathbf{e}_i)_{i=1}^n$  is a canonical basis for  $\mathbb{R}^n$ . Note that the exact value of  $h$  is not specified a priori, hence an additional degree of freedom is added to the filter. More details concerning the filtering applications of central difference approximations can be found in [Ito and Xiong, 2000; Nørgaard et al., 2000; Schei, 1997].

### 3. NONLINEAR BAYESIAN FILTERING

---

The central difference approximation of a Gaussian variable with mean  $\bar{\mathbf{x}}$  and covariance  $\mathbf{P} = \mathbf{S}^T \mathbf{S}$  is given by  $2n + 1$  representative points with the corresponding weights:

$$\begin{aligned} \mathbf{x}_0 &= \bar{\mathbf{x}} & \omega_0 &= \frac{h^2 - n}{h^2}, \\ \mathbf{x}_i &= \bar{\mathbf{x}} \pm \mathbf{S}^T h \mathbf{e}_i & \omega_i &= \frac{1}{2h^2}. \end{aligned}$$

By such a definition the weight of the central point  $\omega_{k-1}^0$  can be negative.

The CDF employs the central difference approximation in both the prediction and the update steps of the filtering algorithm. The complete CDF is presented in Algorithm 3.5.

---

#### Algorithm 3.5 Central Difference Filter

---

**Input:**  $h, \mathbf{P}_{k-1|k-1}, \hat{\mathbf{x}}_{k-1|k-1}$

**Prediction step:**

Factorize the posterior covariance:  $\mathbf{P}_{k-1|k-1} = \mathbf{S}^T \mathbf{S}$

Set the central point:  $\mathbf{x}_{k-1}^0 = \hat{\mathbf{x}}_{k-1|k-1}$

Set the central weight:  $\omega_{k-1}^0 = \frac{h^2 - n}{h^2}$

**for**  $i = 1, \dots, 2n$  **do**

    Compute the representative points:

$$\mathbf{x}_{k-1}^i = \hat{\mathbf{x}}_{k-1|k-1} \pm \mathbf{S}^T h \mathbf{e}_i$$

    Assign weights:  $\omega_{k-1}^i = \frac{1}{2h^2}$

**end for**

Compute the predicted mean  $\hat{\mathbf{x}}_{k|k-1}$  according to (3.40)

Compute the predicted covariance  $\mathbf{P}_{k|k-1}$  according to (3.41)

**Update step:**

Factorize the predicted covariance:  $\mathbf{P}_{k|k-1} = \tilde{\mathbf{S}}^T \tilde{\mathbf{S}}$

Set the central point:  $\mathbf{x}_{k|k-1}^0 = \hat{\mathbf{x}}_{k|k-1}$

Set the central weight:  $\omega_{k|k-1}^0 = \frac{h^2 - n}{h^2}$

**for**  $i = 1, \dots, 2n$  **do**

    Compute the representative points:

$$\mathbf{x}_{k|k-1}^i = \hat{\mathbf{x}}_{k|k-1} \pm \tilde{\mathbf{S}}^T h \mathbf{e}_i$$

    Assign weights:  $\omega_{k|k-1}^i = \frac{1}{2h^2}$

**end for**

Compute the estimated measurement  $\hat{\mathbf{y}}_{k|k-1}$  using (3.42)

Compute the covariance of the predicted observation  $\mathbf{P}_{yy}$  using (3.50)

Compute the cross-covariance of the predicted observation and the predicted state  $\mathbf{P}_{xy}$  using (3.45)

Compute the Kalman gain  $\mathbf{K}_k$  using (3.47)

Compute the estimated mean  $\hat{\mathbf{x}}_{k|k}$  using (3.43)

Compute the estimated covariance  $\mathbf{P}_{k|k}$  using (3.44)

---

When the parameter  $h$  is chosen to be small the central-difference approximation is based on points that are close to the center (mean). When  $h$  is large the approximation accounts for the points located at the tails of the Gaussian distribution.

#### Example: Prediction Step

To illustrate the advantages that the LRKF filters have over the EKF we will use an example of the nonlinear noise-free process given by [Lefebvre et al., 2004]:

$$\mathbf{x}_{k+1}(1) = (x_k(1))^2, \quad (3.58a)$$

$$\mathbf{x}_{k+1}(2) = x_k(1) + 3x_k(2). \quad (3.58b)$$

To better see the differences between the EKF, the UKF, the CDF and the GHF we focus only on a one step ahead prediction problem. The analysis of the update step follows the same steps [Lefebvre et al., 2004].

Assume that at time step  $k$  the state  $\mathbf{x}_k$  is normally distributed with mean  $\mathbf{x}_{k|k} = [10 \ 15]^T$  and covariance  $\mathbf{P}_{k|k} = \begin{bmatrix} 36 & 0 \\ 0 & 3600 \end{bmatrix}$ . We want to predict the distribution of the state  $\mathbf{x}_{k+1|k}$ . The linearization method that is employed by the EKF yields:

$$\hat{\mathbf{x}}_{k+1|k}^{EKF} = \begin{bmatrix} 100 \\ 55 \end{bmatrix}, \mathbf{P}_{k+1|k}^{EKF} = \begin{bmatrix} 14400 & 720 \\ 720 & 32436 \end{bmatrix}.$$

The UKF with parameter  $\lambda = 1$  approximates the distribution of  $\mathbf{x}_{k+1|k}$  with five points:

$$\begin{bmatrix} 100 \\ 55 \end{bmatrix}, \begin{bmatrix} 416 \\ 65 \end{bmatrix}, \begin{bmatrix} 0 \\ 45 \end{bmatrix}, \begin{bmatrix} 100 \\ 367 \end{bmatrix}, \begin{bmatrix} 100 \\ -257 \end{bmatrix}$$

weighted  $1/3, 1/6, 1/6, 1/6,$  and  $1/6$ , respectively. Thus, the mean and covariance of the UKF estimate of the state  $\mathbf{x}_{k+1|k}$  are given by

$$\hat{\mathbf{x}}_{k+1|k}^{UKF} = \begin{bmatrix} 136 \\ 55 \end{bmatrix}, \mathbf{P}_{k+1|k}^{UKF} = \begin{bmatrix} 16992 & 720 \\ 720 & 32436 \end{bmatrix}.$$

For the CDF there are also five representative points, e.g., for  $h = 2$  we have

$$\begin{bmatrix} 100 \\ 55 \end{bmatrix}, \begin{bmatrix} 484 \\ 67 \end{bmatrix}, \begin{bmatrix} 4 \\ 43 \end{bmatrix}, \begin{bmatrix} 100 \\ 415 \end{bmatrix}, \begin{bmatrix} 100 \\ -305 \end{bmatrix}$$

weighted  $1/2, 1/4, 1/4, 1/4,$  and  $1/4$ , respectively. From these the mean and variance are computed

$$\hat{\mathbf{x}}_{k+1|k}^{CDF} = \begin{bmatrix} 136 \\ 55 \end{bmatrix}, \mathbf{P}_{k+1|k}^{CDF} = \begin{bmatrix} 18288 & 720 \\ 720 & 32436 \end{bmatrix}.$$

The number of representative points utilized by the GHF depends on the quadrature order  $m$ . The smallest feasible order is  $m = 2$  which yields the quadrature rule  $\{(x_l, \omega_l)\} = \{(1, 1/2), (-1, 1/2)\}$  from which the representative points are computed:

$$\begin{bmatrix} 16 \\ -131 \end{bmatrix}, \begin{bmatrix} 16 \\ 229 \end{bmatrix}, \begin{bmatrix} 256 \\ -119 \end{bmatrix}, \begin{bmatrix} 256 \\ 241 \end{bmatrix}$$

weighted  $1/4, 1/4, 1/4,$  and  $1/4$ , respectively.

$$\hat{\mathbf{x}}_{k+1|k}^{GHF} = \begin{bmatrix} 136 \\ 55 \end{bmatrix}, \mathbf{P}_{k+1|k}^{GHF} = \begin{bmatrix} 14400 & 720 \\ 720 & 32436 \end{bmatrix}.$$

### 3. NONLINEAR BAYESIAN FILTERING

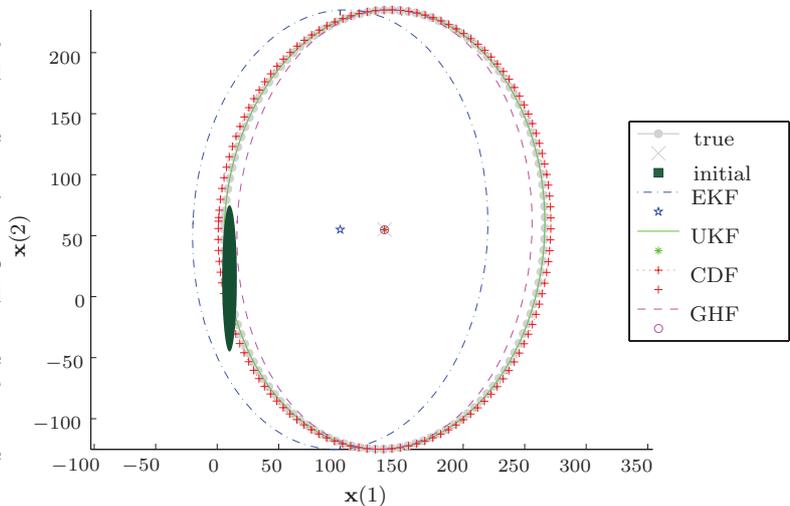
Instead of analyzing algebraic properties of all the covariance matrices  $\mathbf{P}_{k+1|k}$  obtained by each filter it is convenient to look at the error ellipses that they yield:

$$(\mathbf{x} - \hat{\mathbf{x}}_{k+1|k})^T (\mathbf{P}_{k+1|k})^{-1} (\mathbf{x} - \hat{\mathbf{x}}_{k+1|k}) = 1. \quad (3.59)$$

The estimated means and error ellipses obtained by the EKF, the UKF with  $\lambda = 1$ , the CDF with  $h = 1/2$  and the GHF with  $m = 2$  are all compared in Figure 3.4. As a reference, which is labelled as “true” error set, we use the error ellipse defined by the sample mean and sample covariance obtained from  $10^6$  Monte Carlo experiments. These are given by

$$\hat{\mathbf{x}}_{k+1|k}^{MC} = \begin{bmatrix} 136 \\ 55 \end{bmatrix}, \mathbf{P}_{k+1|k}^{MC} = \begin{bmatrix} 17039 & 757 \\ 757 & 32436 \end{bmatrix}.$$

Figure 3.4: Error ellipses of the EKF (dot-dashed line), the UKF (solid line), the CDF (crosses) and the GHF (dashed line) compared with the “true” error ellipse (filled circles) obtained by  $10^6$  Monte Carlo experiments. The shaded area denotes the initial covariance. The means of the UKF (asterisk), the CDF (cross) and the GHF (circle) coincide with the true mean (large x).



As can be observed, only the EKF yields an estimated mean that does not coincide with the “true mean”. The UKF, the CDF and the GHF all provide accurate estimates of the “true” mean but the error ellipses that they produce are different. With such a choice of the parameters ( $\lambda = 1$  for the UKF,  $h = 1/2$  for the CDF,  $m = 2$  for the GHF) the error ellipse obtained by the UKF is the closest to the true one. However, with different parameter setting we can tune the error ellipses to more desirable shapes. In general, by decreasing the value of the parameter  $h$  we will shrink the ellipses  $\mathbf{P}_{k+1|k}^{CDF}$ , and by increasing the quadrature order  $m$  we are able to expand the error ellipses  $\mathbf{P}_{k+1|k}^{GHF}$ . Representative points produced by the UKF, the CDF and the GHF together with the corresponding error ellipses for two parameter settings are presented in Figure 3.5.

#### 3.3.3 Gaussian Sum Filter

So far the discussion has been restricted to systems with Gaussian process and measurement noises. Although this type of stochasticity is most commonly used in modeling real-life processes, in a number of situations one has to deal with non-Gaussian random variables that influence the process or the measurement model [Bilik and Tabrikian, 2006]. The filters discussed in Sections 3.2.3, 3.3.1, and 3.3.2 assume Gaussian noises, hence, when this assumption is violated they no longer perform as expected. Furthermore, even if the noises are Gaussian, the nonlinearities of the state model  $\mathbf{f}_k$  and the observation model  $\mathbf{h}_k$  might produce densities  $p(\mathbf{x}_k | \mathcal{Y}_{k-1})$

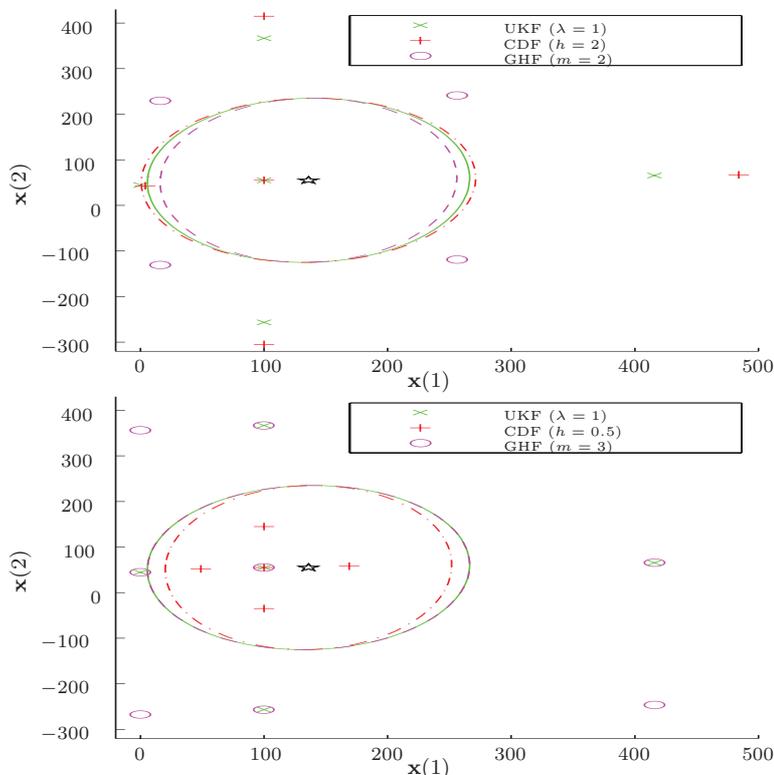


Figure 3.5: Representative points of the UKF ( $\times$ ), the CDF (crosses) and the GHF (circles) and the error ellipses that correspond to covariance matrices obtained by the UKF (solid line), the CDF (dashed-dotted line) and the GHF (dashed line) for  $\lambda = 1, h = 2, m = 2$  (above) and  $\lambda = 1, h = 1/2, m = 3$  (below). Regardless of the parameters' setting all filters yield the same mean (pentagram).

or  $p(\mathbf{x}_k | \mathcal{Y}_k)$  that cannot be accurately approximated by a single normal variable. A possible solution to these problems is the *Gaussian Sum Filter* (GSF) that is described in this section.

### GSF Algorithm

The GSF is based on the theoretical result that an arbitrary probability distribution  $p(\mathbf{x})$  can be approximated by a density  $p_A^N(\mathbf{x})$  of a form:

$$p_A^N(\mathbf{x}) = \sum_{i=1}^N a^i \mathcal{N}(\mathbf{x}; \mu^i, \Sigma^i), \quad (3.60)$$

where for each  $1 \leq i \leq N$ ,  $\mathcal{N}(\mathbf{x}; \mu^i, \Sigma^i)$  is a probability density of a normal distribution, with the mean  $\mu^i$  and the covariance  $\Sigma^i$ , evaluated at  $\mathbf{x}$ , and  $a^i$  are nonnegative weights that sum up to one. The density  $p_A^N(\mathbf{x})$  uniformly converges to the original density  $p(\mathbf{x})$  as the number of terms  $N$  increases and each covariance matrix  $\Sigma_i$  approaches the zero matrix (see [Alspach and Sorenson, 1972; Arasaratnam et al., 2007], and the references therein). Before the *Gaussian Sum* (GS) approximation can be used, one has to specify the parameters  $a^i, \mu^i, \Sigma^i$ . These are usually given as solutions of a certain optimization algorithm. The choice of the optimization method is not trivial, and in general depends on the particular estimation problem. Different methods for parameter selection are discussed in [Alspach and Sorenson, 1971]. Another approach to the problem, which is based on expectation-maximization algorithms is derived in [Verbeek et al., 2006].

As always, there is a tradeoff between computational complexity and the accuracy of the approximation. If one uses too many terms in the summations, the computational time will

### 3. NONLINEAR BAYESIAN FILTERING

---

increase and the filter will no longer be feasible for online applications. On the other hand, if there are too few terms in the GS, the algorithm will produce a poor approximation of the true densities.

The main feature of the GSF is the use of the GS approximation of both the predicted state density  $p(\mathbf{x}_k|\mathcal{Y}_{k-1})$  and the posterior density  $p(\mathbf{x}_k|\mathcal{Y}_k)$ . At each time step  $k$  the aforementioned densities are assumed to be given by:

$$p(\mathbf{x}_k|\mathcal{Y}_{k-1}) = \sum_{i=1}^{N_1} a_{k|k-1}^i \mathcal{N}\left(\mathbf{x}_k; \mu_{k|k-1}^i, \Sigma_{k|k-1}^i\right), \quad (3.61)$$

$$p(\mathbf{x}_k|\mathcal{Y}_k) = \sum_{i=1}^{N_2} a_{k|k}^i \mathcal{N}\left(\mathbf{x}_k; \mu_{k|k}^i, \Sigma_{k|k}^i\right). \quad (3.62)$$

As it was stated before, there is much flexibility in choosing the weights  $a^i$  and the Gaussian parameters  $\mu^i$  and  $\Sigma^i$ . Note that in general the number of terms  $N_1$  in (3.61) does not have to be equal to the number of terms  $N_2$  in (3.62).

One might consider the GSF as a collection of nonlinear Kalman Filters, such as the ones described in the previous sections, working in parallel. Indeed, in the original formulation of Alspach and Sorensen [Alspach and Sorensen, 1971], the GSF that they derived is composed of parallel EKFs. A GSF that exploits UKFs is presented in [Vermaak et al., 2005], whereas a GHF-based GSF can be found in [Arasaratnam et al., 2007].

The filtering proceeds as follows. Let us assume that at time step  $k-1$  the posterior density  $p(\mathbf{x}_{k-1}|\mathcal{Y}_{k-1})$  is represented as a sum of Gaussian densities, i.e.,

$$p(\mathbf{x}_{k-1}|\mathcal{Y}_{k-1}) = \sum_{i=1}^K \alpha_{k-1|k-1}^i \mathcal{N}\left(\mathbf{x}_{k-1}; \hat{\mathbf{x}}_{k-1|k-1}^i, \mathbf{P}_{k-1|k-1}^i\right), \quad (3.63)$$

where  $\alpha_{k-1|k-1}^i$  are non-negative weights that sum up to one, and  $\hat{\mathbf{x}}_{k-1|k-1}^i$  and  $\mathbf{P}_{k-1|k-1}^i$  are the  $i$ -th estimate of the mean and the covariance, respectively. Furthermore, let us also approximate the state noise  $\mathbf{v}_k$  by a GS:

$$p(\mathbf{v}_k) = \sum_{j=1}^L \alpha_{v,k}^j \mathcal{N}\left(\mathbf{v}_k; \hat{\mathbf{v}}_k^j, \mathbf{P}_{v,k}^j\right), \quad (3.64)$$

with weights  $\alpha_{v,k}^j$ , means  $\hat{\mathbf{v}}_k^j$ , and covariances  $\mathbf{P}_{v,k}^j$  chosen to match the non-Gaussian random variable  $\mathbf{v}_k$ . Then for each pair  $(i, j)$ ,  $i = 1, \dots, K$ ,  $j = 1, \dots, L$  the  $(i, j)$ -th component of the predicted state density is computed by the nonlinear KF of one's choice. The predicted state density is thus given by:

$$p(\mathbf{x}_k|\mathcal{Y}_{k-1}) = \sum_{i=1}^K \sum_{j=1}^L \alpha_{k|k-1}^{i,j} \mathcal{N}\left(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}^{i,j}, \mathbf{P}_{k|k-1}^{i,j}\right), \quad (3.65)$$

where the weights  $\alpha_{k|k-1}^{i,j}$  are computed as:

$$\alpha_{k|k-1}^{i,j} = \alpha_{k-1|k-1}^i \alpha_{v,k}^j, \quad (3.66)$$

and  $\hat{\mathbf{x}}_{k|k-1}^{i,j}$  and  $\mathbf{P}_{k|k-1}^{i,j}$  are estimates of the mean and the covariance, respectively, that are obtained by the application of one of the filters described in Sections 3.3.1 and 3.3.2 to the

### 3.3. Parametric Nonlinear Bayesian Filtering

model with index  $(i, j)$ . To perform the update step, we again use the GS to approximate the observation noise  $\mathbf{w}_k$ :

$$p(\mathbf{w}_k) = \sum_{l=1}^M \alpha_{w,k}^l \mathcal{N}(\mathbf{w}_k; \hat{\mathbf{w}}_k^l, \mathbf{P}_{w,k}^l). \quad (3.67)$$

Next, for each tuple  $(i, j, l)$ ,  $i = 1, \dots, K, j = 1, \dots, L, l = 1, \dots, M$  the update step is performed by a nonlinear KF of one's choice (EKF, UKF, GHF, etc.). Finally, the separate steps are combined, resulting in the posterior density:

$$p(\mathbf{x}_k | \mathcal{Y}_k) = \sum_{i=1}^K \sum_{j=1}^L \sum_{l=1}^M \alpha_{k|k}^{i,j,l} \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}^{i,j,l}, \mathbf{P}_{k|k}^{i,j,l}), \quad (3.68)$$

where the weights  $\alpha_{k|k}^{i,j,l}$  are given by:

$$\alpha_{k|k}^{i,j,l} = \frac{\alpha_{k|k-1}^{i,j} \alpha_{w,k}^l p_{i,j,l}(\mathbf{y}_k | \mathcal{Y}_{k-1})}{\sum_{i=1}^K \sum_{j=1}^L \sum_{l=1}^M \alpha_{k|k-1}^{i,j} \cdot \alpha_{w,k}^l p_{i,j,l}(\mathbf{y}_k | \mathcal{Y}_{k-1})}, \quad (3.69)$$

and the mean  $\hat{\mathbf{x}}_{k|k}^{i,j,l}$  and the covariance  $\mathbf{P}_{k|k}^{i,j,l}$  are obtained from the chosen nonlinear KF applied separately to each tuple  $(i, j, l)$ . In the above formula the term  $p_{i,j,l}(\mathbf{y}_k | \mathcal{Y}_{k-1})$  denotes the  $(i, j, l)$ -th component of a PDF of observing  $\mathbf{y}_k$  at step  $k$  given the past observations  $\mathcal{Y}_{k-1}$ , which can be approximated by the Gaussian:

$$p_{i,j,l}(\mathbf{y}_k | \mathcal{Y}_{k-1}) = \mathcal{N}(\mathbf{y}_k; \hat{\mathbf{y}}_{k|k-1}^{i,j,l}, \mathbf{P}_{yy}^{i,j,l}). \quad (3.70)$$

Algorithm 3.6 summarizes the GSF that applies the UKF (the sigma points  $\mathbf{x}_\sigma$  and  $\omega_\sigma$  are computed according to Algorithm 3.3) to each component of the GS approximation in both the prediction and in the update stage. Note that if one replaces the UKF with an other nonlinear filter, e.g., the EKF, the general structure of Algorithm 3.6 remains intact. Indeed, the two algorithms are different only in the formulas for the means:  $\hat{\mathbf{x}}_{k|k-1}, \hat{\mathbf{x}}_{k|k}, \hat{\mathbf{y}}_{k|k-1}$  and the covariances:  $\mathbf{P}_{k|k-1}, \mathbf{P}_{k|k}, \mathbf{P}_{yy}$ .

#### Reduction Methods

In the general framework presented in Section 3.3.3, at the beginning of the algorithm there are  $K$  components in the summation, whereas the final number of terms to sum up is  $KLM$ . At the next filtering step the algorithm starts with  $KLM$  initial expressions, and hence it finishes with  $KL^2M^2$ . After  $k$  steps there are  $KL^kM^k$  terms to sum up. This means that as the filtering proceeds, the number of the expressions in the summation grows exponentially. Therefore, in its basic form, the GSF has a very limited practical use.

To overcome this potential drawback of the GSF, several techniques have been developed to reduce the number of terms in the GS approximations [Ali-Löytty, 2008b; Arasaratnam et al., 2007; Horwood and Poore, 2011; Ito and Xiong, 2000; Tam et al., 1999; Terejanu et al., 2008]. Among the popular methods are:

**1. Pruning:** In this approach the mixture components with negligible weights are discarded from the GS, whereas the remaining terms have the weights uniformly rescaled so that the GS forms a probability density function. Depending on the problem, one might discard every component which has the weight smaller than a fixed threshold  $\epsilon$  or terms that have the cumulative weight smaller than  $\epsilon$  [Arasaratnam et al., 2007].

### 3. NONLINEAR BAYESIAN FILTERING

---

**Algorithm 3.6** Gaussian Sum Filter as a collection of UKFs

---

**Input:**  $\left\{ \left( \alpha_{k-1|k-1}^i, \hat{\mathbf{x}}_{k-1|k-1}^i, \mathbf{P}_{k-1|k-1}^i \right) \right\}_{i=1}^K, \left\{ \left( \alpha_{v,k}^j, \hat{\mathbf{v}}_k^j, \mathbf{P}_{v,k}^j \right) \right\}_{j=1}^L, \left\{ \left( \alpha_{w,k}^l, \hat{\mathbf{w}}_k^l, \mathbf{P}_{w,k}^l \right) \right\}_{l=1}^M$

**Prediction step:**

**for**  $i = 1, \dots, K, j = 1, \dots, L$  **do**

    Compute the predicted mean:

$$\hat{\mathbf{x}}_{k|k-1}^{i,j} = \sum_{\sigma} \omega_{\sigma}^{i,j} \mathbf{f}_k(\mathbf{x}_{\sigma}^{i,j})$$

    Compute the predicted covariance:

$$\mathbf{P}_{k|k-1}^{i,j} = \sum_{\sigma} \omega_{\sigma}^{i,j} \left( \mathbf{f}_k(\mathbf{x}_{\sigma}^{i,j}) - \hat{\mathbf{x}}_{k|k-1}^{i,j} \right) \left( \mathbf{f}_k(\mathbf{x}_{\sigma}^{i,j}) - \hat{\mathbf{x}}_{k|k-1}^{i,j} \right)^T$$

    Compute the associated weight:

$$\alpha_{k|k-1}^{i,j} = \alpha_{k-1|k-1}^i \alpha_{v,k}^j$$

**end for**

Approximate the predicted state density with the Gaussian Sum:

$$p(\mathbf{x}_k | \mathcal{Y}_{k-1}) = \sum_{i=1}^K \sum_{j=1}^L \alpha_{k|k-1}^{i,j} \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}^{i,j}, \mathbf{P}_{k|k-1}^{i,j})$$

**Update state:**

**for**  $i = 1, \dots, K, j = 1, \dots, L, l = 1, \dots, M$  **do**

    Compute the mean of the predicted observation:

$$\hat{\mathbf{y}}_{k|k-1}^{i,j,l} = \sum_{\sigma} \omega_{\sigma}^{i,j,l} \mathbf{h}_k(\mathbf{x}_{\sigma}^{i,j,l})$$

    Compute the covariance of the predicted observation:

$$\mathbf{P}_{yy}^{i,j,l} = \sum_{\sigma} \omega_{\sigma}^{i,j,l} \left( \mathbf{h}_k(\mathbf{x}_{\sigma}^{i,j,l}) - \hat{\mathbf{y}}_{k|k-1}^{i,j,l} \right) \left( \mathbf{h}_k(\mathbf{x}_{\sigma}^{i,j,l}) - \hat{\mathbf{y}}_{k|k-1}^{i,j,l} \right)^T$$

    Compute the cross-covariance of the predicted observation and the predicted state:

$$\mathbf{P}_{xy}^{i,j,l} = \sum_{\sigma} \omega_{\sigma}^{i,j,l} \left( \mathbf{x}_{\sigma}^{i,j,l} - \hat{\mathbf{x}}_{k|k-1}^{i,j} \right) \left( \mathbf{h}_k(\mathbf{x}_{\sigma}^{i,j,l}) - \hat{\mathbf{y}}_{k|k-1}^{i,j,l} \right)^T$$

    Use (3.43)-(3.47) to compute the updated mean:

$$\hat{\mathbf{x}}_{k|k}^{i,j,l} = \hat{\mathbf{x}}_{k|k-1}^{i,j} + \mathbf{K}_k^{i,j,l} (\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}^{i,j,l})$$

    Use (3.44)-(3.47) to compute the updated covariance:

$$\mathbf{P}_{k|k}^{i,j,l} = \mathbf{P}_{k|k-1}^{i,j} - \mathbf{P}_{xy}^{i,j,l} \left( \mathbf{K}_k^{i,j,l} \right)^T$$

    Compute the associated weight:

$$\alpha_{k|k}^{i,j,l} = \frac{\alpha_{k|k-1}^{i,j} \alpha_{w,k}^l \mathcal{N}(\mathbf{y}_k; \hat{\mathbf{y}}_{k|k-1}^{i,j,l}, \mathbf{P}_{yy}^{i,j,l})}{\sum_{i=1}^K \sum_{j=1}^L \sum_{l=1}^M \alpha_{k|k-1}^{i,j} \alpha_{w,k}^l \mathcal{N}(\mathbf{y}_k; \hat{\mathbf{y}}_{k|k-1}^{i,j,l}, \mathbf{P}_{yy}^{i,j,l})}$$

**end for**

Approximate the posterior density with the Gaussian Sum:

$$p(\mathbf{x}_k | \mathcal{Y}_k) = \sum_{i=1}^K \sum_{j=1}^L \sum_{l=1}^M \alpha_{k|k}^{i,j,l} \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}^{i,j,l}, \mathbf{P}_{k|k}^{i,j,l})$$


---

### 3.3. Parametric Nonlinear Bayesian Filtering

**2. Merging:** When using this method one joins the Gaussian densities that are close to each other with respect to a certain distance, namely the Mahalanobis distance [Arasaratnam et al., 2007; Horwood and Poore, 2011; Tam et al., 1999; Williams, 2003]:

$$d_{ij}^2 = \frac{\alpha^i \alpha^j}{\alpha^i + \alpha^j} (\hat{\mathbf{x}}^i - \hat{\mathbf{x}}^j)^T (\mathbf{P}^i + \mathbf{P}^j)^{-1} (\hat{\mathbf{x}}^i - \hat{\mathbf{x}}^j). \quad (3.71)$$

This algorithm in general merges mixture terms that have lower weights rather than those that are associated to higher weights [Arasaratnam et al., 2007].

The GS approximations obtained by pruning or merging procedure converge weakly to the exact posterior distribution [Ali-Löytty, 2008b].

**3. Integral Squared Error-Based Gaussian Mixture Reduction:** In this method one obtains the reduced Gaussian mixture expressions by minimizing the  $L_2$  distance between the original and the reduced densities [Arasaratnam et al., 2007; Ito and Xiong, 2000]:

$$\operatorname{argmin}_{\alpha, \boldsymbol{\mu}, \boldsymbol{\Sigma}, N} \int \left( p(\mathbf{x}_k | \mathcal{Y}_k) - \sum_{i=1}^N \alpha^i \mathcal{N}(\mathbf{x}_k; \boldsymbol{\mu}^i, \boldsymbol{\Sigma}^i) \right)^2 d\mathbf{x}_k \quad (3.72)$$

where  $p(\mathbf{x}_k | \mathcal{Y}_k)$  is the original Gaussian Sum approximation defined by (3.68),  $N$  is the desired number of components in the Gaussian mixture that is usually much smaller than the number of terms in original GS, and  $\alpha, \boldsymbol{\mu}, \boldsymbol{\Sigma}$  are the parameters with respect to which the optimization is performed. In some cases instead of the  $L_2$  distance other metrics are used as the optimization criterion, e.g., see [Williams, 2003].

Using one of the aforementioned techniques one has control over the number of terms in the GS, and hence the growing memory requirement ceases to be a problem. However, the reduction procedure, which can be computationally very expensive, has to be performed at each filtering step. Therefore, depending on the problem, an appropriate choice of the reduction method is crucial to make the GSF an effective online filter. Note that if both the process and the observation noises can be accurately approximated by single Gaussians, no reduction method is necessary because the number of expressions in the GS is constant over the time.

#### Example: Kinematic Model

Let us consider a second order kinematic model in two-dimensional space [Farina et al., 2002; Li and Jilkov, 2003; Zarchan, 2002]. The model is described by four states:

$$\mathbf{x}_k = \begin{bmatrix} p_x(k) \\ \dot{p}_x(k) \\ p_y(k) \\ \dot{p}_y(k) \end{bmatrix}, \quad (3.73)$$

where  $(p_x(k), p_y(k))$  is the position of an object at time  $k$  in the  $XY$  plane and  $\dot{p}_x(k)$  and  $\dot{p}_y(k)$  denote the velocity of the object at time  $k$  in the  $X$ -direction and the  $Y$ -direction, respectively. The evolution of the object in discrete-time is modeled by:

$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & T_s & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} \frac{T_s^2}{2} & 0 \\ \frac{T_s}{T_s} & 0 \\ 0 & \frac{T_s^2}{2} \\ 0 & T_s \end{bmatrix} \left( \mathbf{f}(\mathbf{x}_k) - \begin{bmatrix} 0 \\ g \end{bmatrix} \right) + \mathbf{v}_k. \quad (3.74)$$

### 3. NONLINEAR BAYESIAN FILTERING

The function  $\mathbf{f}(\mathbf{x}_k)$  is given by

$$\mathbf{f}(\mathbf{x}_k) = -0.5 \frac{g}{\beta} \rho(\mathbf{x}_k(3)) \sqrt{(\mathbf{x}_k(2))^2 + (\mathbf{x}_k(4))^2} \begin{bmatrix} \mathbf{x}_k(2) \\ \mathbf{x}_k(4) \end{bmatrix}, \quad (3.75)$$

with parameters  $T_s = 1[\text{s}]$  the sampling time,  $g = 9.81[\text{m/s}^2]$  the gravitational acceleration,  $\beta = 100[\text{kg/m}^2]$  the ballistic coefficient,  $\rho(\mathbf{x}_k(3)) = 1.754 \cdot \exp(-1.491 \cdot \mathbf{x}_k(3))$  the air density (typically modeled as an exponentially decaying function of height [Farina et al., 2002]). Furthermore, the variable  $\mathbf{v}_k$  models the process noise, which is a zero-mean Gaussian with the covariance matrix  $\mathbf{Q}_k$  equal to

$$\mathbf{Q}_k = \begin{bmatrix} 33\frac{1}{3} & 50 & 0 & 0 \\ 50 & 100 & 0 & 0 \\ 0 & 0 & 33\frac{1}{3} & 50 \\ 0 & 0 & 50 & 100 \end{bmatrix}. \quad (3.76)$$

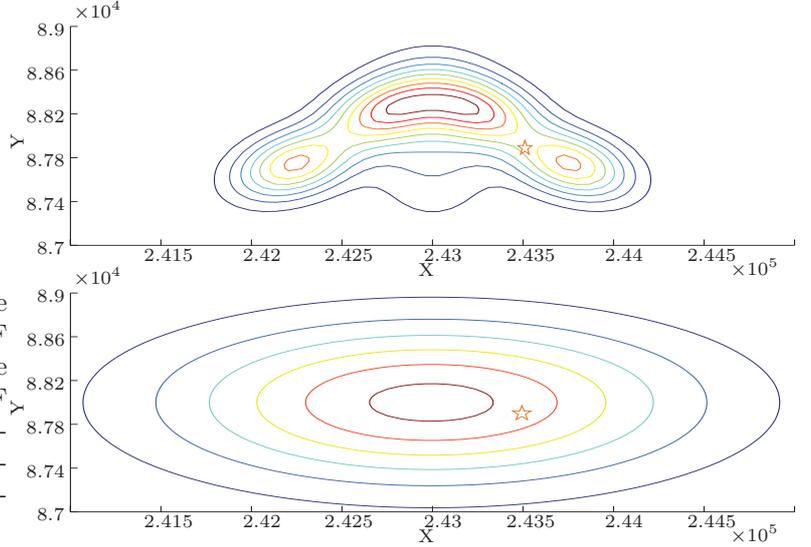


Figure 3.6: Contour of the initial PDF of the GSF (above) vs contour of the initial PDF of the UKF (below) both in the XY position plane. The pentagram denotes the true initial state of the system.

For the observation model we assume that at each time step  $k$  the range  $\mathbf{y}_k(1)$  and bearing  $\mathbf{y}_k(2)$  measurements are available [Farina et al., 2002; Li and Jilkov, 2003]. Thus, in the cartesian coordinates the measurement model is given by:

$$\mathbf{y}_k(1) = \sqrt{(\mathbf{x}_k(1))^2 + (\mathbf{x}_k(3))^2} + \mathbf{w}_k(1), \quad (3.77a)$$

$$\mathbf{y}_k(2) = \arctan\left(\frac{\mathbf{x}_k(3)}{\mathbf{x}_k(1)}\right) + \mathbf{w}_k(2), \quad (3.77b)$$

where the zero-mean Gaussian variable  $\mathbf{w}_k$  models the random measurement noise with covariance matrix  $\mathbf{R}_k = \begin{bmatrix} 10^4 & 0 \\ 0 & 0.01 \end{bmatrix}$ . With such a choice of  $\mathbf{R}_k$  the standard deviation of the range errors is equal to  $\sigma_r = 100[\text{m}]$  and the standard deviation in bearing errors is given by  $\sigma_\theta = 0.1[\text{rad}]$ .

We have simulated a trajectory of the ballistic object for  $T = 90$  time steps, which corresponds to  $90\text{s}$ , starting from the initial state  $\mathbf{x}_0 = [243.5\text{km}, 1000\text{m/s}, 87.9\text{km}, 0\text{m/s}]^T$ . The simulation was repeated 1000 times.

### 3.3. Parametric Nonlinear Bayesian Filtering

We use the Monte Carlo experiment described above to compare the performance of a five-term GSF with a UKF. The initial condition  $\mathbf{x}_0^{UKF}$  and initial covariance  $\mathbf{P}_0^{UKF}$  for the UKF are given by:

$$\mathbf{x}_0^{UKF} = \begin{bmatrix} 243 \cdot 10^3 \\ 1000 \\ 88 \cdot 10^3 \\ 0 \end{bmatrix}, \quad \mathbf{P}_0^{UKF} = \begin{bmatrix} 10^6 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 25 \cdot 10^4 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}.$$

The initial condition for the GSF is given by five equally weighted Gaussians with means  $\mu_i, i = 1, \dots, 5$ :

$$\begin{bmatrix} 242250 \\ 1000 \\ 87750 \\ 0 \end{bmatrix}, \begin{bmatrix} 242750 \\ 1000 \\ 88250 \\ 0 \end{bmatrix}, \begin{bmatrix} 243250 \\ 1000 \\ 88250 \\ 0 \end{bmatrix}, \begin{bmatrix} 243750 \\ 1000 \\ 87750 \\ 0 \end{bmatrix}, \begin{bmatrix} 243000 \\ 1000 \\ 88000 \\ 0 \end{bmatrix},$$

respectively, and the covariances

$$\Sigma_1 = \Sigma_2 = \begin{bmatrix} \frac{3 \cdot 250^2}{4} & 0 & \frac{250^2}{4} & 0 \\ 0 & 100 & 0 & 0 \\ \frac{250^2}{4} & 0 & \frac{3 \cdot 250^2}{4} & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}, \Sigma_3 = \Sigma_4 = \begin{bmatrix} \frac{3 \cdot 250^2}{4} & 0 & -\frac{250^2}{4} & 0 \\ 0 & 100 & 0 & 0 \\ -\frac{250^2}{4} & 0 & \frac{3 \cdot 250^2}{4} & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix},$$

$$\Sigma_5 = \begin{bmatrix} 250^2 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 500^2 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}.$$

The contour of the initial distribution of the GSF is visualized in Figure 3.6. The initial PDF has been chosen to resemble the parabolic shape of the trajectory of the ballistic object. Such a shape cannot be achieved by a single Gaussian distribution.

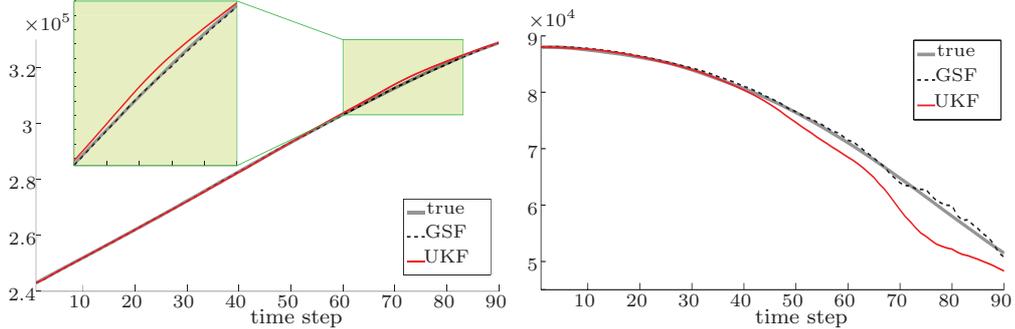


Figure 3.7: Tracking of the X-position (left) and Y-position (right) of the ballistic object (thick solid line) by the GSF (dashed line) and the UKF (thin solid line).

Figure 3.7 presents the simulated XY-trajectory of the ballistic object together with the estimates obtained by the GSF and the UKF. It can be easily observed that the GSF outperforms a single UKF. This is confirmed by the analysis of the RMSE of each filter obtained from 1000

### 3. NONLINEAR BAYESIAN FILTERING

Monte Carlo runs of the system (3.74)–(3.77) with the same initial condition and the same noise levels. In Figure 3.7 the RMSE of the GSF and the UKF are compared with the squared root of the theoretical PCRB that is computed using (3.38).

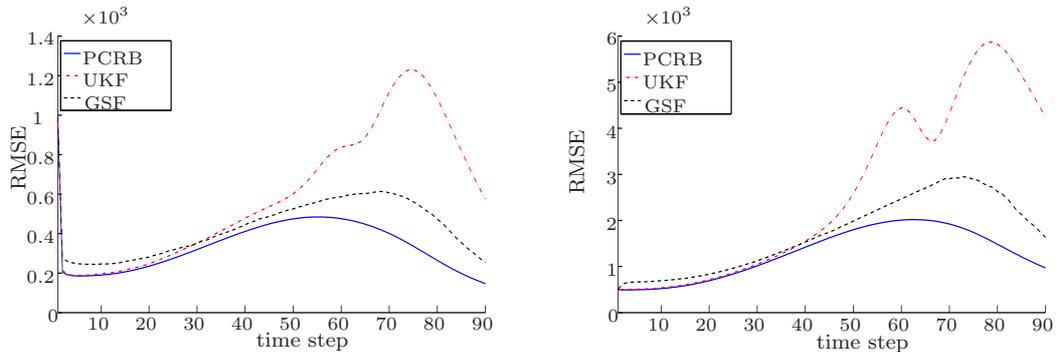


Figure 3.8: RMSE of the GSF (dashed line) and the UKF (dashed-dotted line) compared with the square root of the theoretical PCRB (solid line) for X-position (left) and Y-position (right).

## 3.4 Nonparametric Nonlinear Bayesian Filtering

In this section we discuss another type of nonlinear filters: Nonparametric Bayesian Filters. Within this framework we no longer assume that the PDF (3.5), which is a solution to the estimation problem, belongs to any particular class of distributions (Gaussian, Sum of Gaussians, etc.). The need for such an approach arose from the fact that in many applications, especially when dealing with highly nonlinear dynamics, the methods described in Section 3.3 fail to give accurate approximations of the true PDF, which can be of arbitrary shape. The nonparametric methods described in what follows are, in general, capable of matching any type of probability distribution. On the down side, in most cases, they are numerically more involved than the parametric methods as the estimation problem becomes infinite dimensional.

In what follows we discuss the Particle Filter (PF) that is a nonparametric method based on the Monte Carlo approach. We focus on two approaches, one originating from the importance sampling method and an other derived from mean-field control approach.

### 3.4.1 Generic Particle Filter

We start by explaining how the general PF works. In order to match the standard notions that appear in the PF literature sometimes we use a slightly different notation than the one used in Section 3.3.

#### Notation

From the model (3.1)–(3.2) we derive the *transition probability kernel*  $K_{k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$  defined by

$$K_{k-1}(\mathbf{x}_k|\mathbf{x}_{k-1}) := \mathbb{P}_w(\mathbf{f}_{k-1}(\mathbf{x}_{k-1}, \mathbf{w}_{k-1}) = \mathbf{x}_k), \quad (3.78)$$

### 3.4. Nonparametric Nonlinear Bayesian Filtering

---

i.e., the conditional PDF of the variable  $\mathbf{x}_k$  given the previous state  $\mathbf{x}_{k-1}$ , and the *likelihood function*  $g_k(\mathbf{y}_k|\mathbf{x}_k)$  defined by

$$g_k(\mathbf{y}_k|\mathbf{x}_k) := \mathbb{P}_v(\mathbf{h}_k(\mathbf{x}_k, \mathbf{v}_k) = \mathbf{y}_k), \quad (3.79)$$

i.e., the conditional PDF of the variable  $\mathbf{y}_k$  given the current state  $\mathbf{x}_k$ .

The true posterior PDF of the state  $\mathbf{x}_k$  given the observations  $\mathcal{Y}_k$  is denoted by  $\pi_{k|k}$ , i.e.,

$$\pi_{k|k} := p(\mathbf{x}_k|\mathcal{Y}_k). \quad (3.80)$$

#### Nonparametric Approximation

The PF represents the posterior PDF  $\pi_{k|k}$  of the state  $\mathbf{x}_k$  by  $N$  random samples (particles)  $\{\mathbf{x}_k^i\}_{i=1}^N$  with their associated weights  $\{\omega_k^i\}_{i=1}^N$ , normalized so that  $\sum_{i=1}^N \omega_k^i = 1$ . At time instant  $k$ , the previous posterior PDF  $\pi_{k-1|k-1}$  is represented by  $N$  samples  $\{\mathbf{x}_{k-1}^i\}_{i=1}^N$  and the corresponding weights  $\{\omega_{k-1}^i\}_{i=1}^N$ . The posterior PDF  $\pi_{k|k}$  is represented by the set of weighted samples, conventionally denoted by:

$$\pi_{k|k} \approx \pi_{k|k}^N := \sum_{i=1}^N \omega_k^i \delta(\mathbf{x}_k - \mathbf{x}_k^i), \quad (3.81)$$

where  $\delta$  denotes the Dirac delta function.

The tuple  $\{(\mathbf{x}_k^i, \omega_k^i)\}_{i=1}^N$  is further used to approximate functionals of form:

$$\mathbb{E}_{\pi_{k|k}} = (f(\mathbf{x}_k)) \int f(\mathbf{x}_k) \pi_{k|k}(\mathbf{x}_k|\mathcal{Y}_k) d\mathbf{x}_k, \quad (3.82)$$

by

$$I^{MC}(f) = \sum_{i=1}^N \omega_k^i f(\mathbf{x}_k^i) \delta(\mathbf{x}_k - \mathbf{x}_k^i), \quad (3.83)$$

where  $f$  is a sufficiently regular function of the state  $\mathbf{x}_k$ .

#### 3.4.2 Sequential Importance Sampling

To approximate the true posterior  $\pi_{k|k}$ , new samples  $\{\mathbf{x}_k^i\}_{i=1}^N$  and weights  $\{\omega_k^i\}_{i=1}^N$  are generated using the *Sequential Importance Sampling* (SIS) method [Arulampalam et al., 2002; Doucet et al., 2000]. The SIS method is a recursive algorithm that uses the most recent observation  $\mathbf{y}_k$  to compute  $\{(\mathbf{x}_k^i, \omega_k^i)\}_{i=1}^N$  in two steps. First, for every  $i = 1, \dots, N$ , the sample  $\mathbf{x}_k^i$  is drawn from a (chosen) *importance kernel*  $\tilde{K}_{k-1}(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i, \mathbf{y}_k)$ . Next, using the most recent observation  $\mathbf{y}_k$ , the weights  $\omega_k^i$  are updated according to the Bayes rule

$$\tilde{\omega}_k^i = \omega_{k-1}^i \frac{g_k(\mathbf{y}_k|\mathbf{x}_k^i) K_{k-1}(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)}{\tilde{K}_{k-1}(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i, \mathbf{y}_k)} \quad (3.84)$$

and normalized

$$\omega_k^i = \frac{\tilde{\omega}_k^i}{\sum_{j=1}^N \tilde{\omega}_k^j}. \quad (3.85)$$

### 3. NONLINEAR BAYESIAN FILTERING

---

#### Importance Sampling

In theory, the best possible importance density is the posterior PDF  $\pi_{k|k}$  itself. In such a case the estimator of the integral (3.82) is unbiased and the variance of the approximation error is of order  $\mathcal{O}(1/N)$  [Kleptsyna and Veretennikov, 2011]. For every other choice of the importance density the variance of the weights  $\{\omega_k^i\}_{i=1}^N$  is greater. With a “proper” choice of the importance density it is possible to achieve asymptotically unbiased estimators of the integral (3.82) with the convergence rate of the variance of the approximation error being again equal to  $\mathcal{O}(1/N)$  [Kleptsyna and Veretennikov, 2011]. Nevertheless, the variance of weights  $\{\omega_k^i\}_{i=1}^N$  increases over time [Doucet et al., 2000]. Since it is generally impossible to sample from the posterior PDF  $\pi_{k|k}$  one needs to rely on suboptimal importance densities in practice. It has been shown (see [Doucet et al., 2000] and the references therein) that the importance density that minimizes the variance of the weights  $\{\omega_k^i\}_{i=1}^N$  conditional upon simulated trajectories  $\{\mathbf{x}_j\}_{j=1}^{k-1}$  and the observations  $\{\mathbf{y}_j\}_{j=1}^k$  is equal to  $\mathbb{P}(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{y}_k)$ , i.e., the PDF of the state  $\mathbf{x}_k$  conditioned upon the immediately preceding realization  $\mathbf{x}_{k-1}^i$  of the simulated trajectory and the most current observation  $\mathbf{y}_k$ .  $\mathbb{P}(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{y}_k)$  is an optimal importance density in the aforementioned sense. Unfortunately, apart from a restricted class of systems [Doucet et al., 2000; Kotecha and Djurić, 2003a,b], sampling from  $\mathbb{P}(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{y}_k)$  is practically impossible. Therefore, various suboptimal importance densities have been proposed during the recent years. The simplest choice is to use an importance density that is fixed over the time [Tanizaki and Mariano, 1994]. A more popular approach is utilized in the *bootstrap particle filter* (BPF) [Gordon et al., 1993; Kitagawa, 1996]. The BPF samples points  $\{\mathbf{w}_{k-1}^i\}_{i=1}^N$  from the noise distribution of  $\mathbf{w}_{k-1}$ , then propagates them together with the particles  $\{\mathbf{x}_{k-1}^i\}_{i=1}^N$ , which approximate the previous posterior PDF  $\pi_{k-1|k-1}$ , through the prediction model (5.1a) to obtain the particles  $\mathbf{x}_k^i := \mathbf{f}_{k-1}(\mathbf{x}_{k-1}^i, \mathbf{w}_{k-1}^i)$  that approximate the predicted PDF  $\pi_{k|k-1}$ . A slightly different formulation of the BPF, with particles  $\{\mathbf{x}_{k-1}^i\}_{i=1}^N$  sampled directly from the transition probability  $\mathbb{P}(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$ , is presented in [Tanizaki and Mariano, 1998]. Another variation of the BPF [Cristian and Doucet, 2002] allows sampling particles  $\{\mathbf{x}_{k-1}^i\}_{i=1}^N$  from the weighted transition probability  $\frac{1}{N} \sum_{j=1}^N \mathbb{P}(\mathbf{x}_k|\mathbf{x}_{k-1}^j)$ .

#### Resampling

A common problem of PF is the particle degeneracy: after several iterations, all but few particles will have negligible weights. This does not come as a surprise since the variance of the weights  $\{\omega_k^i\}_{i=1}^N$  can only increase over time. When that occurs most of the computational power is wasted on updating negligible weights and the accuracy of the algorithm strongly deteriorates since the posterior PDF  $\pi_{k|k}$  is approximated only by a small set of significant particles. The degeneracy phenomenon can be circumvented by monitoring the weights and resampling the particles, e.g., with Algorithm 3.7, after the degeneracy is detected [Fu and Jia, 2010; Kong et al., 1994; Lee and Chia, 2002; Ristic et al., 2004a]. A common measure of the degeneracy is the *effective sample size*  $N_{\text{eff}}$ , computed by [Liu and Chen, 1998; Ristic et al., 2004a]:

$$N_{\text{eff}} = \frac{1}{\sum_{i=1}^N (\omega_k^i)^2}. \quad (3.86)$$

---

### 3.4. Nonparametric Nonlinear Bayesian Filtering

---

Alternatively one can test a Kullback-Leibler distance between the sets of weights obtained in the consecutive iterations [Lee and Chia, 2002], or simply measure how much mass is concentrated on the particle with the maximum weight at each iteration before resampling [Fu and Jia, 2010; Lee and Chia, 2002].

---

#### Algorithm 3.7 Resampling

---

**Output:**  $\{(\mathbf{x}_{\text{new}}^i, \omega_{\text{new}}^i)\}_{i=1}^N$   
**for**  $i = 1, 2, \dots, N$  **do**  
    Compute cumulative sum of weights:  $\omega_c^i = \sum_{j=1}^i \omega_k^j$   
**end for**  
Draw  $u_1$  from  $\mathcal{U}(0, \frac{1}{N})$   
**for**  $i = 1, 2, \dots, N$  **do**  
    Find  $\mathbf{x}^{+i}$ , the first sample such that  $\omega_c^i \geq u_i$ .  
    Replace particle  $i$ :  $\mathbf{x}_{\text{new}}^i = \mathbf{x}^{+i}$ ,  $\omega_{\text{new}}^i = \frac{1}{N}$   
     $u_{i+1} = u_i + \frac{1}{N}$   
**end for**

---

The PF that monitors the degeneracy using  $N_{\text{eff}}$  is summarized in Algorithm 3.8. More information on importance sampling resampling algorithms can be found in [Carpenter et al., 1999; Hurzeler and Kunsch, 1998; Künsch, 2005; Pitt and Shephard, 1999].

---

#### Algorithm 3.8 Particle filter

---

**Input:**  $K_{k-1}(\mathbf{x}_k | \mathbf{x}_{k-1})$ ,  $\tilde{K}_{k-1}(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{y}_k)$ ,  $g_k(\mathbf{y}_k | \mathbf{x}_k)$ ,  $p_0(\mathbf{x}_0)$ ,  $N$ ,  $N_T$   
**Initialize:**  
**for**  $i = 1, 2, \dots, N$  **do**  
    Draw a new particle:  $\mathbf{x}_0^i \sim p(\mathbf{x}_0)$   
    Assign weight:  $\omega_0^i = \frac{1}{N}$   
**end for**  
**At every time step**  $k = 1, 2, 3, \dots$   
**for**  $i = 1, 2, \dots, N$  **do**  
    Draw particle from importance distribution:  
 $\mathbf{x}_k^i \sim \tilde{K}_{k-1}(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i, \mathbf{y}_k)$   
    Use measured  $\mathbf{y}_k$  to update the weight:  
 $\tilde{\omega}_k^i = \omega_{k-1}^i \frac{g_k(\mathbf{y}_k | \mathbf{x}_k^i) K_{k-1}(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i)}{\tilde{K}_{k-1}(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i, \mathbf{y}_k)}$   
**end for**  
Normalize weights:  $\omega_k^i = \frac{\tilde{\omega}_k^i}{\sum_{j=1}^N \tilde{\omega}_k^j}$   
**if**  $\frac{1}{\sum_{i=1}^N (\omega_k^i)^2} < N_T$  **then**  
    Resample using Algorithm 3.7.  
**end if**

---

#### Asymptotic Convergence of the Generic Particle Filter

The analysis of the convergence of the PF-approximation  $\pi_{k|k}^N$  to the true posterior PDF  $\pi_{k|k}$  requires the use of advanced concepts from the measure theory. This means that the PF presented in previous sections needs to be redefined into somewhat more abstract form. The discussion regarding the asymptotic properties of the PF requires much attention and is carried

### 3. NONLINEAR BAYESIAN FILTERING

---

out separately in Chapter 6. Here we only mention that it is possible to prove that under certain conditions the empirical distribution  $\pi_{k|k}^N$  converges almost surely to the true posterior PDF  $\pi_{k|k}$ . A comprehensive overview of the convergence properties can be found in [Cristian and Doucet, 2002; Doucet et al., 2001].

#### 3.4.3 Mean-Field Control-Oriented Approach

In previous section we have discussed how the PF, which is derived from the importance sampling principle, works. In this section we present an alternative approach to nonlinear filtering which originates from the mean-field optimal control techniques [Huang et al., 2007; Yin et al., 2010]. Recently, based on this principle the *Feedback Particle Filter* (FPF) has been derived [Yang et al., 2011a,b]. The FPF has been developed for the continuous-time stochastic dynamical systems, thus, before we proceed to the description of the FPF algorithm we give the continuous-time formulation of the system.

##### Continuous-Time Dynamical Systems

The continuous-time analog of the discrete-time dynamical system (3.1)–(3.2) is given by [Budhiraja et al., 2007; Øksendal, 2003b; Yang et al., 2011a,b]

$$d\mathbf{x}_t = \mathbf{f}(\mathbf{x}_t) dt + d\mathbf{v}_t, \quad (3.87a)$$

$$d\mathbf{y}_t = \mathbf{h}(\mathbf{x}_t) dt + d\mathbf{w}_t, \quad (3.87b)$$

where  $\mathbf{x}_t \in \mathbb{R}^n$  and  $\mathbf{y}_t \in \mathbb{R}^p$  denote the state and the observation processes and  $\mathbf{v}_t, \mathbf{w}_t$  are mutually independent standard Wiener processes with stochastic volatilities  $\sigma_v$  and  $\sigma_w$ , respectively.

Note that in real-life problems the measurements are in general collected at discrete-time intervals. Thus the continuous-time observation model should be considered as the limit case of the discrete-time observation model. Such a formulation is convenient in deriving the FPF.

##### Feedback Particle Filter

The evolution of the conditional probability density  $\pi(\mathbf{x}, t) = \mathbb{P}(\mathbf{x}_t | \mathbf{Y}_t)$  is described by the Kushner-Stratonovich equation [Yang et al., 2011b, 2012]:

$$d\pi = \left( -\nabla(\pi\mathbf{f}) + \frac{1}{2}\Delta\pi \right) dt + \left( \mathbf{h} - \hat{\mathbf{h}} \right) \left( d\mathbf{y}_t - \hat{\mathbf{h}}dt \right) \pi, \quad (3.88)$$

where  $\nabla$  is gradient operator,  $\Delta$  is the Laplacian operator, and  $\hat{\mathbf{h}}$  denotes the expectation of the observation model:

$$\hat{\mathbf{h}} := \int \mathbf{h}(\mathbf{x}) \pi(\mathbf{x}, t) d\mathbf{x}. \quad (3.89)$$

The aim of the FPF is to force particles  $\mathbf{x}_t^i$  to mimic the evolution of the true state in the probabilistic sense. In other words the objective of the FPF is to match the true PDF  $\pi(\mathbf{x}, t)$  evolving according to (3.88) by the PDFs of the particles  $\mathbf{x}_t^i$ . This is achieved by designing an appropriate control law such that for the particles  $\mathbf{x}_t^i$  with the initial distribution  $p(\mathbf{x}, 0)$  equal to the initial state of the system  $\pi(\mathbf{x}, 0)$  the distributions remain equivalent for all times  $t$ , i.e.,  $p(\mathbf{x}, t) = \pi(\mathbf{x}, t)$  for all  $t \geq 0$ .

The dynamics of each particle is described by the following *Stochastic Differential Equation* (SDE) [Yang et al., 2011b, 2012]:

$$d\mathbf{x}_t^i = \left( \mathbf{f}(\mathbf{x}_t^i) + u(\mathbf{x}_t^i, t) \right) dt + \mathbf{K}(\mathbf{x}_t^i, t) d\mathbf{y}_t + d\mathbf{v}_t^i, \quad (3.90)$$

### 3.4. Nonparametric Nonlinear Bayesian Filtering

---

where  $\{u, \mathbf{K}\}$  are control actions that need to be determined by the filter. Furthermore, we assume that  $\{\mathbf{x}_0^i\}_{i=1}^N$  - the initial distributions of the particles are i.i.d., with the distribution equal to  $\pi(\mathbf{x}, 0)$  and mutually independent of the process noises  $\{\mathbf{v}_t^i\}_{i=1}^N$ .

It has been proven in [Yang et al., 2011b, 2012] that, under certain regularity conditions, the control actions  $\{u, \mathbf{K}\}$  that force the particles to follow, in probabilistic sense, the desired true state distribution  $\pi$  can be obtained in two steps:

- I. First, to obtain the gain function  $\mathbf{K}$  we need to solve the multidimensional Euler-Lagrange boundary value problem given by:

$$\nabla(p(\mathbf{x}, t) \nabla \phi(\mathbf{x}, t)) = -(\mathbf{h}(\mathbf{x}) - \hat{\mathbf{h}}) p(\mathbf{x}, t), \quad (3.91a)$$

$$\int \phi(\mathbf{x}, t) p(\mathbf{x}, t) d\mathbf{x} = 0, \quad (3.91b)$$

where  $p(\mathbf{x}, t)$  is the conditional distribution of  $\mathbf{x}_t^i$  given  $\mathcal{Y}_t$ . Given  $\phi(\mathbf{x}, t)$  - the solution of (3.91), the gain function  $\mathbf{K}$  is computed as:

$$[\mathbf{K}]_{ls}(\mathbf{x}, t) := \frac{\partial \phi_s}{\partial \mathbf{x}_l}(\mathbf{x}, t). \quad (3.92)$$

- II. Secondly, the control action  $u$  is given by the following formula:

$$u(\mathbf{x}, t) = -\frac{1}{2} \mathbf{K}(\mathbf{x}, t) (\mathbf{h}(\mathbf{x}) - \hat{\mathbf{h}}^N) + \Omega(\mathbf{x}, t), \quad (3.93)$$

where  $\hat{\mathbf{h}}^N$  is a sample mean that approximates the expectation of the observation model (3.89):

$$\hat{\mathbf{h}}^N := \frac{1}{N} \sum_{i=1}^N \mathbf{h}(\mathbf{x}_t^i), \quad (3.94)$$

and  $\Omega = (\Omega_1, \dots, \Omega_n)$  is the Wong-Zakai correction term:

$$\Omega_l(\mathbf{x}, t) = \frac{1}{2} \sum_{k=1}^n \sum_{s=1}^p \mathbf{K}_{ks}(\mathbf{x}, t) \frac{\partial \mathbf{K}_{ls}}{\partial \mathbf{x}}(\mathbf{x}, t). \quad (3.95)$$

In [Yang et al., 2011b, 2012] it has been proven that the control actions  $\{u, \mathbf{K}\}$  defined above are the unique ones that achieve the formulated objective.

The particles  $\mathbf{x}_t^i$  are coupled through the term  $\hat{\mathbf{h}}^N$  which combines information provided by all the particles. It has been argued that using this amount of global information is sufficient to improve the performance of PFs [Yang et al., 2011a]. The important feature of the FPF is that it avoids resampling and weighs all the particles equally. It has been observed that the FPF-estimates are of lower variance than the estimates obtained by the BPF [Yang et al., 2011b]. Furthermore, the control-oriented approach provides a self-correcting feedback loop that has stabilizing effect on the particles  $\mathbf{x}_k^i$ .

From the numerical perspective the biggest drawback of the FPF is the need to solve the Euler-Lagrange boundary value problem at every time step. This is further complicated by the fact that the solution of the partial differential equation given by (3.91a) requires the knowledge of the distribution  $p(\mathbf{x}, t)$ . This, however is rarely the case in nonlinear problems, which leaves one to rely on approximations, e.g., Gaussian Sum approximation [Yang et al., 2011b, 2012]. In general terms, the problem of computing the gain function  $\mathbf{K}(\mathbf{x}, t)$  as a solution to (3.91) is related to the problem of finding the solution to the Hamilton-Jacobi-Bellman equations. Thus one can draw from a rich literature dealing with the subject of stochastic control [Oksendal, 2003a].

## 3.5 Conclusions and Discussion

The main objective of this section is to analyze the properties of nonlinear filters presented in the previous sections. For general nonlinear non-Gaussian systems, there exists no optimal solution to the filtering problem (in the MMSE sense). This means that there are no results stating that a particular filter has the lowest possible MMSE error [Ristic et al., 2004a].

We have presented five types of nonlinear filtering methods:

- I. Parametric Filters based on analytical approximations: EKF, IEKF.
- II. Parametric Filters based on statistical approximations: UKF, GHF, CDF.
- III. Parametric Filters based on Gaussian Sum approximations: GSF.
- IV. Nonparametric Filters based on the importance sampling approach: BPF.
- V. Nonparametric Filters based on the mean-field control-oriented approach: FPF.

The EKF, the IEKF, the UKF, the GHF, and the CDF approximate the predicted (3.4) and the posterior (3.5) densities as Gaussians. The EKF and the IEKF utilize the Taylor series expansion to exploit the analytical structure of nonlinear functions  $\mathbf{f}_k$  and  $\mathbf{h}_k$ . The UKF, the GHF and the CDF exploit statistical properties of Gaussian variables that undergo nonlinear transformations. In contrast to the aforementioned methods the GSF approximates the densities (3.4) and (3.5) with the sum of Gaussian densities, which are no longer Gaussian. The biggest advantage of nonparametric filters is that they do not assume any particular shape of the predicted and the posterior distributions and thus are capable of approximating PDFs of arbitrary shapes.

The Taylor approximation, which is the basic principle of the EKF and the IEKF, requires the functions  $\mathbf{f}_k$  and  $\mathbf{h}_k$  to be differentiable. The UKF, the GHF, and the CDF are derivative-free filters, i.e., they can be applied to systems with non-differentiable dynamics. The same applies to the BPF, the FPF, and the GSF if it uses one of the derivative-free methods.

The numerical complexity of the UKF and the CDF grows linearly with the dimension of the state  $n$ , the numerical complexity of the EKF and the IEKF grows quadratically with  $n$ , and the complexity of the GHF grows exponentially with  $n$ . In the case of the GSF there is no straightforward relation between the dimension of the state space  $n$  and the computational complexity of the filter. The latter depends on the number of terms  $K$  in the GS that are required for an accurate approximation of the densities (3.4) and (3.5). In general, a larger  $K$  is necessary for higher dimensions [Julier and Uhlmann, 1996], but the exact relation always depends on the particular structure of the approximated densities. For the nonparametric BPF and FPF there is also no straightforward relation between the dimension of the state space and the computational time required by the method. It is generally accepted that the higher the dimension of the state space the more particles are needed to obtain a good approximation of the true posterior PDF. Nevertheless, in recent years there have been many successful applications of the Ensemble Particle Filters, which are PFs using relatively small number of samples, to high dimensional estimation problems [Evensen, 2003, 2006; Kim et al., 2003].

The Taylor series approximation truncates the higher moments of nonlinear function. Therefore, filters derived from this principle, such as the EKF, are better suited for systems where the functions  $\mathbf{f}_k$  and  $\mathbf{h}_k$  are mildly nonlinear. From this perspective the advantage of the UKF, the GHF and the CDF over the EKF is that these filters match higher-order moments and thus, can handle stronger nonlinearities in the system equations. Among these three filters the UKF has the simplest form and while being more accurate than the EKF, it retains its low computational complexity. The CDF, though similar to the UKF, is able to estimate the

state covariance more precisely. This, however, comes with the price of increased computational complexity. The GHF, when using a sufficiently large quadrature rule, is able to accurately approximate heavy tailed distributions. The disadvantage of the GHF over the EKF, the CDF and the UKF is its numerical complexity which often yields the GHF impractical for high-frequency online applications.

The performance of the EKF can be improved by using the measurement to minimize linearization errors. This is achieved by the IEKF, the trade-off being the increase of the numerical complexity.

The performance of the EKF, the IEKF, the UKF, the GHF or the CDF can deteriorate if the predicted and the posterior densities cannot be accurately approximated by a single Gaussian. If the system exhibits severely non-Gaussian characteristics, the GSF offers a neat alternative to the aforementioned filters.

The GHF, and to a lesser degree the EKF and the IEKF, suffers from the curse of dimensionality. Therefore, from the computational perspective, the GHF, the EKF, and the IEKF are better suited for small-scale systems, whereas the UKFs and the CDFs are more suitable for large-scale applications. Whenever the nonlinear functions  $\mathbf{f}_k$  or  $\mathbf{h}_k$  have complicated analytical structures, which make it difficult to compute the Jacobians  $\partial\mathbf{f}_k$  or  $\partial\mathbf{h}_k$ , the derivative-free filters (UKF, CDF, GHF) are numerically preferable over the EKF.

The BPF is based on the Monte Carlo importance sampling approach which means that at each filtering step multiple samples need to be obtained from the importance distribution. This might be computationally expensive if the importance distribution is not well chosen. Another computational burden comes from the resampling step of the filter, which is necessary to avoid sample degeneracy. Nevertheless, despite these problems the BPF is an extremely popular filtering tool with many new modifications being developed every year, usually tackling one of the aforementioned problems.

The FPF avoids the resampling step that is embedded into the SIS PF as all the particles are equally weighted  $\omega_k^i = \frac{1}{N}$ . Furthermore, the control-oriented formulation provides a self-correcting feedback loop that stabilizes the particles  $\mathbf{x}_k^i$  around the common posterior  $\pi(\mathbf{x}, t)$ . On the down side, to implement the FPF one needs to solve a certain Euler-Lagrange boundary value problem at each time-step. Feasible methods for overcoming this obstacle are still under development [Yang et al., 2011b] as this problem is still a matter of ongoing research.

We would like to conclude the chapter with a brief overview of freely available implementations of the algorithms discussed throughout the chapter. Mathworks provides the MatLab codes for the EKF [Mat, a], the UKF [Mat, b] and the PF [mat]. A very useful overview of open source MatLab and C++ toolboxes used for nonlinear filtering, including KF, EKF, UKF and PF, is provided by Greg Welch and Gary Bishop [Welch and Bishop]. A comprehensive collection of MatLab toolboxes suited for nonlinear filtering, among others EKF, UKF, CDF, GSF, PF is provided by the Identification and Decision Making Research Group at the University of West Bohemia [Ide].

### 3. NONLINEAR BAYESIAN FILTERING

---

## Chapter 4

# Solutions to the Drag-Head Estimation Problems

### Abstract

The Drag-Head is the most important component of the excavation system in a Trailing Suction Hopper Dredger (TSHD). Its task is to break the coherence of the bottom soil which allows the loosened material to be sucked in by the dredge pump and be transported to the hopper. The production comes mostly from cutting and jetting, the processes that are traditionally controlled by the operator. To automate the control of the two processes several soil-dependent parameters need to be estimated from the available measurements. These parameters are: the ratio  $k_{vh}$  between cutting forces, the horizontal cutting force coefficient  $k_{ch}$ , and the in situ permeability  $k_{si}$ . The parameters need to be estimated online due to the time-varying nature of the excavation process. Some of the measurements used by the estimation algorithms are available with a time-varying delay. The method of handling the delay is discussed firstly, independently of the estimation methods. Next, we discuss the solutions to two estimation problems associated with the excavation process: the Cutting Estimation Problem that comes from considering the cutting-only production mode, and the Cutting and Jetting Estimation Problem that originate from complete cutting with the jetting production mode.

### 4.1 Introduction

The Drag-Head is a mechanical tool used by TSHDs to excavate the soil from the bottom of the sea. In Chapter 2 the dynamical excavation models were presented that contain a number of uncertain soil-dependent parameters. Furthermore, the corresponding estimation problems were formulated.

To recapitulate, the objective is to obtain online estimates of the following time varying parameters: the ratio  $k_{vh}$  between the cutting forces, the horizontal cutting force coefficient  $k_{ch}$  and the in situ permeability  $k_{si}$  from the available measurements. These parameters are needed for the automatic controller of the TSHD.

This chapter proposes solutions to these problems. First, we show how to handle the time-varying delay in the measurement of the incoming density  $\rho_i$ , which is equal to the time required by the mixture entering the Drag-Head to reach the sensor located in the transport pipe. Next, in Section 4.3 we provide a solution to the Cutting Estimation Problem. Section 4.4 is devoted

## 4. SOLUTIONS TO THE DRAG-HEAD ESTIMATION PROBLEMS

---

to finding a solution to the Cutting and Jetting Estimation Problem. The chapter is concluded with recommendations and directions of further research.

### 4.2 Handling the Time-Varying Measurement Delay

In this section we derive an algorithm that solves the problem of time-varying delay in the measurements of the incoming density  $\rho_i$  that was introduced in Chapter 2. For the sake of completeness of the section we start by recalling the measurement delay problem.

#### 4.2.1 Measurement Delay in the Drag-Head System

Recall that in the dynamical systems for which the Drag-Head estimation problem is formulated, the measurement of the excavation depth  $h_{ex}$  is assumed to be available. However, this value is calculated from the values of the incoming flow rate  $Q_i$ , incoming density (at the drag-head inlet)  $\rho_i^{dh}$ , the in situ sand density  $\rho_s$ , the ship speed's  $v_{sh}$ , and the fixed values of water density  $\rho_w$  and the width of the Drag-Head  $W_d$  by the formula:

$$h_{ex}(t) = \frac{Q_i(t) (\rho_i^{dh}(t) - \rho_w)}{(\rho_s(t) - \rho_w) W_d v_{sh}(t)}. \quad (4.1)$$

The signals  $Q_i(t)$ ,  $v_{sh}(t)$ , and  $\rho_s(t)$  are assumed to be known at time  $t$  without errors. This is justified by the fact that the variables  $Q_i$  and  $v_{sh}$  are measured onboard the ship whereas the variable  $\rho_s$  depends on the average grain diameter  $d_m$ , the parameter that is obtained by solving the Hopper Estimation Problem which will be discussed in Chapter 7. Thus, we treat them as known inputs. The value of the incoming density at the Drag-Head inlet  $\rho_i^{dh}$  is measured with the transport delay  $\tau_t$  at the pump  $\rho_i^m$ .

At time  $t$ , the measurement of the incoming mixture density  $\rho_i^m(t)$  is obtained at the pump, which is the delayed incoming density at the Drag-Head inlet  $\rho_i^{dh}(t - \tau_t)$ . Hence, we have the relation:

$$\rho_i^m(t) = \rho_i^{dh}(t - \tau_t). \quad (4.2)$$

This transport delay  $\tau_t$  in the measurement of  $\rho_i^{dh}$  equals the time needed by the mixture entering the Drag-Head to reach the sensor located in the pipeline. The delay is time-varying as it depends on the incoming flow rate  $Q_i$ , which is a time-varying variable.

#### Delayed Estimate

The dynamics of the Drag-Head with delayed measurements discussed above are described by a continuous-time dynamical system of type:

$$\dot{\mathbf{x}}_t = \mathbf{f}_t(\mathbf{x}_t, \mathbf{w}_t, \mathbf{u}_t) \quad (4.3)$$

$$\mathbf{y}_{t-\tau_t} = \mathbf{h}_{t-\tau_t}(\mathbf{x}_{t-\tau_t}, \mathbf{v}_{t-\tau_t}, \mathbf{u}_t), \quad (4.4)$$

where one of the input variables, say  $\mathbf{u}_t^1$  is known after a delay  $\tau_t$ . This delay can be calculated from the remaining input signals  $\mathbf{u}_{i-\tau_t:t}^{2:n}$  recorded during the interval  $[t - \tau_t, t]$ . In the case of the Drag-Head model the incoming density at the Drag-Head inlet  $\rho_i^{dh}$  is such a delayed variable.

Then, at time  $t$  the  $\tau_t$ -delayed MMSE estimate of the state  $\mathbf{x}_{t-\tau_t}$  is obtained by a standard Bayesian dynamic filter, which uses information up to time  $t$ :

$$\hat{\mathbf{x}}_{t-\tau_t} = \mathbb{E}(\mathbf{x}_{t-\tau_t} | \mathbf{y}_{0:t-\tau_t}, \mathbf{u}_{0:t}). \quad (4.5)$$

### Calculation of the Delay $\tau_t$

In the continuous time model, in principle, at all time  $t$  it is possible to calculate the exact value of the transport delay  $\tau_t$  by solving, in  $\tau_t$ , the volume balance equation

$$\int_{t-\tau_t}^t Q_i(s) ds = \pi \left( \frac{d}{2} \right)^2 l, \quad (4.6)$$

where  $d$  and  $l$  denote the diameter of the pipe (assuming constant pipe diameter) and the length of the pipe, respectively.

In a discrete-time model, in general, obtaining the exact value of  $\tau_t$  is no longer possible. This is because the samples of the incoming flow  $Q_i$  are available in discrete moments  $0, t_s, 2t_s, \dots, kt_s, \dots$  for a given sampling time  $t_s$ . Then, the discrete-time analogue of (4.6) for  $t = nt_s$  is given by

$$\sum_{k: kt_s \in [t-\tau_t, t]} t_s Q_i(kt_s) = \pi \left( \frac{d}{2} \right)^2 l. \quad (4.7)$$

### Discrete-Time Solution

Unfortunately, except for some special cases, (4.7) has no solution, which means that the delay  $\tau_t$  is not a multiple of the sampling time  $t_s$ . However, we can always approximate  $\tau_t$  by  $\hat{\tau}_t$  with  $t_s$ -accuracy i.e., the delay approximation error is bounded by sampling time:

$$|\hat{\tau}_t - \tau_t| \leq t_s. \quad (4.8)$$

Such an approximation is described in Algorithm 4.1.

---

#### Algorithm 4.1 Estimate of the delay

---

**Input:**  $t = mt_s, \{Q_i(0), \dots, Q_i(t)\}$

Compute  $k_0$  such that the following conditions are satisfied:

$$\sum_{t_k = t - k_0 t_s}^t t_s Q_i(t_k) \geq \pi \left( \frac{d}{2} \right)^2 l, \quad (4.9a)$$

$$\sum_{t_k = t - (k_0 - 1)t_s}^t t_s Q_i(t_k) < \pi \left( \frac{d}{2} \right)^2 l. \quad (4.9b)$$

Compute the approximated delay  $\tau_t \in ((k_0 - 1)t_s, t_s]$  by

$$\hat{\tau}_t := k_0 t_s, \quad (4.10)$$


---

### 4.2.2 Measurement Synchronization

Before the procedure described in previous section can be successfully implemented two practical problems need to be tackled:

- I. Non uniqueness of  $\rho_i^{dh}$ ,

#### 4. SOLUTIONS TO THE DRAG-HEAD ESTIMATION PROBLEMS

---

##### II. Missing measurements of $\rho_i^{dh}$ .

Both of these problems arise as a consequence of the discrete nature of the considered signals. To explain these phenomena let us consider the following. Assume that for times  $t_{m-1} = (m-1)t_s$  and  $t_m = mt_s$  we have computed the corresponding delays  $\hat{\tau}_{t_{m-1}}$  and  $\hat{\tau}_{t_m}$  respectively. These delays can be correlated in three possible ways that will be discussed in this section.

##### Perfect Synchronization: $\hat{\tau}_{t_m} = \hat{\tau}_{t_{m-1}}$

If  $\hat{\tau}_{t_m} = \hat{\tau}_{t_{m-1}}$  holds, the algorithm is perfectly synchronized. Indeed, first, the value  $\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}})$  is used to compute the estimate  $\hat{\mathbf{x}}(t_{m-1} - \hat{\tau}_{t_{m-1}})$ . To obtain the state estimate at the consecutive time step  $t_{m-1} - \hat{\tau}_{t_{m-1}} + t_s$  we need the value of  $\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}} + t_s)$ . But, since  $\hat{\tau}_{t_m} = \hat{\tau}_{t_{m-1}}$  and  $t_m = t_{m-1} + t_s$ , we see that  $\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}} + t_s)$  equals to  $\rho_i^{dh}(t_m - \hat{\tau}_{t_m})$ , which, by definition, is equal to the consecutive measurement  $\rho_i^m(t_{m-1} + t_s)$ . Thus, for the consecutive times  $t_{m-1}, t_{m-1} + t_s$  we obtain consecutive state estimates  $\hat{\mathbf{x}}(t_{m-1} - \hat{\tau}_{t_{m-1}}), \hat{\mathbf{x}}(t_{m-1} - \hat{\tau}_{t_{m-1}} + t_s)$ .

The situation is a bit more complicated if  $\hat{\tau}_{t_m} \neq \hat{\tau}_{t_{m-1}}$ . We discuss this case in the following sections starting from the case of  $\hat{\tau}_{t_m} > \hat{\tau}_{t_{m-1}}$ .

##### Non Unique Measurement: $\hat{\tau}_{t_m} > \hat{\tau}_{t_{m-1}}$

Consider the case of  $\hat{\tau}_{t_m} > \hat{\tau}_{t_{m-1}}$ . This is likely to occur when the incoming flow rate  $Q_i$  decreased between the sampling times  $t_{m-1}$  and  $t_m$ . Note that by the definition of  $\hat{\tau}_t$  we have

$$\hat{\tau}_{t+t_s} \leq \hat{\tau}_t + t_s. \quad (4.11)$$

This combined with  $\hat{\tau}_{t_m} > \hat{\tau}_{t_{m-1}}$  yields:  $\hat{\tau}_{t_m} = \hat{\tau}_{t_{m-1}} + t_s$ . Hence, we have:

$$\begin{aligned} \rho_i^{dh}(t_m - \hat{\tau}_{t_m}) &= \rho_i^{dh}(t_{m-1} + t_s - \hat{\tau}_{t_{m-1}} - t_s) \\ &= \rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}}) \\ &= \rho_i^m(t_{m-1}) \end{aligned}$$

But, by the definition of  $\rho_i^{dh}(t - \hat{\tau}_t)$ , we also get

$$\rho_i^{dh}(t_m - \hat{\tau}_{t_m}) := \rho_i^m(t_m).$$

Consequently, we have

$$\begin{aligned} \rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}}) &= \rho_i^m(t_{m-1}), \\ \rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}}) &= \rho_i^m(t_m), \end{aligned}$$

thus, at time  $t_{m-1} - \hat{\tau}_{t_{m-1}}$  the value of  $\rho_i^{dh}$  is not uniquely defined.

This somewhat surprising outcome is the result of discretization. The ‘‘true’’ continuous-time delays  $\tau_{t_{m-1}}, \tau_{t_m}$  are such that

$$t_{m-1} - \hat{\tau}_{t_{m-1}} \leq t_{m-1} - \tau_{t_{m-1}} < t_m - \tau_{t_m} < t_{m-1} - \hat{\tau}_{t_{m-1}} + t_s,$$

which means that both  $t_{m-1} - \tau_{t_{m-1}}$  and  $t_m - \tau_{t_m}$  lay in the same interval and thus the discretization cannot distinguish  $t_{m-1} - \hat{\tau}_{t_{m-1}}$  from  $t_m - \hat{\tau}_{t_m}$ . The simplest way to handle the resulting ambiguity in  $\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}})$  is to choose one value of, e.g., the first that was obtained for a given time  $t_{m-1} - \hat{\tau}_{t_{m-1}}$ . Then

$$\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}}) := \rho_i^m(\min\{t_{m-1}, t_m\}). \quad (4.12)$$

## 4.2. Handling the Time-Varying Measurement Delay

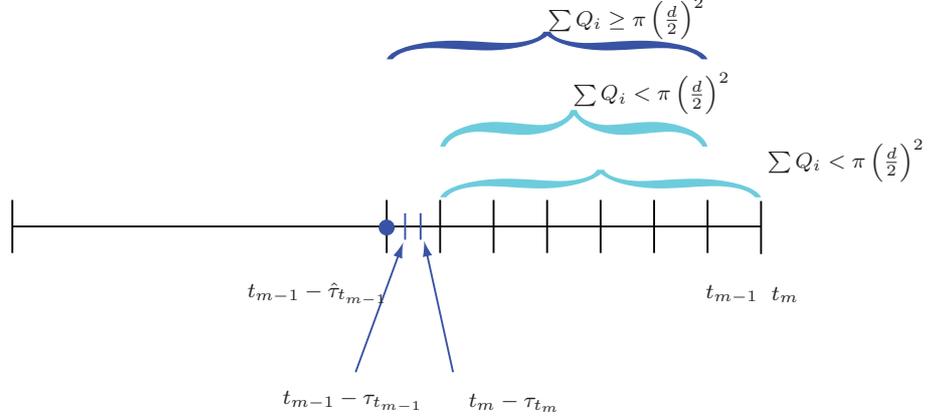


Figure 4.1: Non unique value of  $\rho_i^{dh}$  caused by time discretization.

An other possible solution is to take the mean of the values  $\rho_i^m(t_{m-1})$  and  $\rho_i^m(t_m)$ . The outcomes of these approaches will not differ much if the dynamics of the incoming density  $\rho_i^m$  is slowly varying with respect to the sampling time  $t_s$ . The situation of non unique measurements is depicted in Figure 4.1

**Missing Measurements:**  $\hat{\tau}_{t_m} < \hat{\tau}_{t_{m-1}}$

Finally let us consider the situation of missing measurements. This is likely to occur when the incoming flow rate  $Q_i$  increased between the sampling times  $t_{m-1}$  and  $t_m$ . From  $\hat{\tau}_{t_m} < \hat{\tau}_{t_{m-1}}$  we get

$$t_m - \hat{\tau}_{t_m} \geq t_{m-1} - \hat{\tau}_{t_{m-1}} + 2t_s. \quad (4.13)$$

Without loss of generality we may assume equality in (4.13). Computing the values of  $\rho_i^{dh}$  we obtain

$$\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}}) = \rho_i^m(t_{m-1}) \quad (4.14)$$

$$\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}} + 2t_s) = \rho_i^{dh}(t_m - \hat{\tau}_{t_m}) \quad (4.15)$$

$$= \rho_i^m(t_m) \quad (4.16)$$

As we can see the value of  $\rho_i^{dh}$  at time  $t_{m-1} - \hat{\tau}_{t_{m-1}} + t_s$  is not defined. This is again due to discretization and corresponds to the continuous-time situation where during the time  $t_s$  the value of  $\tau_t$  decreased more than  $2t_s$ , i.e.,  $\tau_t dt|_{t_0}^{t_0+t_s} < -2$ . This is visualized in Figure 4.2.

The missing measurements are problematic for a filter that uses single-step dynamical model of the generic form:

$$\mathbf{x}_{t_k+t_s} = \mathbf{f}(\mathbf{x}_{t_k}, \mathbf{w}_{t_k}, \mathbf{u}_{t_k}) \quad (4.17)$$

$$\mathbf{y}_{t_k-\tau_k} = \mathbf{h}(\mathbf{x}_{t_k-\tau_k}, \mathbf{v}_{t_k-\tau_k}, \mathbf{u}_{t_k}). \quad (4.18)$$

This is because without the observation  $\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}} + t_s)$  the state estimate  $\hat{\mathbf{x}}_{t_k - \hat{\tau}_{t_{m-1}} + t_s}$  cannot be computed by the filter and, due to the iterative character of the Bayesian dynamic filter, none of the following estimates can be obtained. Fortunately, the missing observation  $\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}} + t_s)$  can be easily produced by interpolating between the closest available observations, which in this case are  $\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}})$  and  $\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}} + 2t_s)$ . Then

$$\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}} + t_s) := \frac{1}{2}\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}}) + \frac{1}{2}\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}} + 2t_s) \quad (4.19)$$

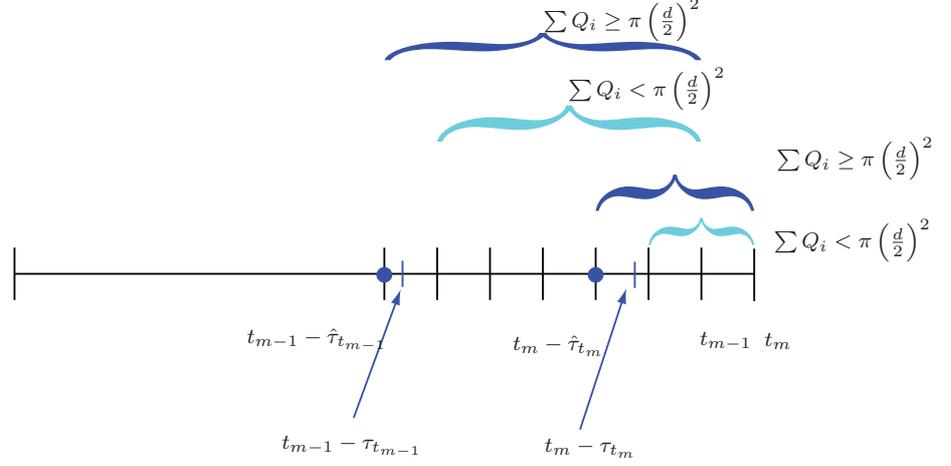


Figure 4.2: Missing measurements between  $\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}})$  and  $\rho_i^{dh}(t_m - \hat{\tau}_{t_m})$  caused by time discretization.

Such an approach is easily generalized to cases where  $|t_m - \hat{\tau}_{t_m} - (t_{m-1} - \hat{\tau}_{t_{m-1}})| > 2t_s$ .

### 4.2.3 Implementation of the Drag-Head Estimation

In this section we show how to relate the discrete-time algorithm handling the measurement delays with the Drag-Head Estimation Problems. The filtering starts at time  $t_0$ , when the first measurement  $\rho_i^m(t_0)$  is collected, with initial delay  $\hat{\tau}_{t_0}$ .

Given the state estimate at time  $t_0 - \hat{\tau}_{t_0}$  we want to compute all the consecutive estimates  $\hat{\mathbf{x}}(t_0 - \hat{\tau}_{t_0} + kt_s)$  for all  $k > 1$ . The iterative method of computing the delayed estimates is derived in Algorithm 4.1.

The method of calculating the delay described in Algorithm 4.1 is independent of the dynamical filter one wants to use to solve the estimation problems. Therefore, in Algorithm 4.2 we do not specify which filter is used to obtain estimates of the states. The choice of the filter depends on the estimation problem under consideration. The block structure of the algorithm is presented in Figure 4.3.

Due to the dynamic nature of the delay, the number of estimates computed at each sample can be 0 (Non unique measurements), 1 (Perfect synchronization) or more (Missing measurements). Due to the latter case, one needs to secure enough computational power to be able to compute several estimates during a single time step.

## 4.3 Solutions to the Cutting Estimation Problem

In this section we derive a discrete-time stochastic dynamical model of the cutting process discussed in Section 2.2. Such a model serves as a basis for the dynamical filters that are used to estimate the ratio  $k_{vh}$  between the cutting forces and the horizontal cutting force coefficient  $k_{ch}$ .

Throughout this section we make the following assumptions:

- I. We assume that the ratio  $k_{vh}$  is a constant parameter for a given soil type but it varies between different soils.

---

### 4.3. Solutions to the Cutting Estimation Problem

---

**Algorithm 4.2** Delayed estimates

---

**Input:**  $t_0, t_s, \{\rho_i^m(t_0 + mt_s)\}_{m \geq 0}$

**Initialization:**

Compute the estimate of the transport delay  $\hat{\tau}_{t_0}$  using Algorithm 4.1

Compute  $\rho_i^{dh}(t_0 - \hat{\tau}_{t_0})$  using (4.2):

$$\rho_i^{dh}(t_0 - \hat{\tau}_{t_0}) = \rho_i^m(t_0)$$

Compute  $h_{ex}(t_0)$  using (4.1)

Combine  $\hat{\mathbf{x}}(0)$  with a nonlinear filter to compute the first delayed estimate  $\hat{\mathbf{x}}(t_0 - \hat{\tau}_{t_0}) = \hat{\mathbf{x}}(t_s)$

**Iteration**  $(m - 1) \rightarrow m$ :

**for**  $t_m = t_0 + mt_s, m \geq 1$  **do**

    Compute the estimate of the transport delay  $\hat{\tau}_{t_m}$  using Algorithm 4.1

**if**  $\hat{\tau}_{t_m} > \hat{\tau}_{t_{m-1}}$  (non unique measurements) **then**

        By (4.12) there is no new input  $\rho_i^{dh}$ , hence no new state estimate is produced

**else if**  $\hat{\tau}_{t_m} = \hat{\tau}_{t_{m-1}}$  (perfect synchronization) **then**

        Compute  $\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}} + t_s)$  using (4.2):

$$\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}} + t_s) = \rho_i^m(t_m)$$

        Compute  $h_{ex}(t_{m-1} - \hat{\tau}_{t_{m-1}} + t_s)$  using (4.1)

        Combine  $\hat{\mathbf{x}}(t_{m-1} - \hat{\tau}_{t_{m-1}})$  with a nonlinear filter to compute the delayed estimate  $\hat{\mathbf{x}}(t_{m-1} - \hat{\tau}_{t_{m-1}} + t_s)$

**else if**  $\hat{\tau}_{t_m} < \hat{\tau}_{t_{m-1}}$  (missing measurements) **then**

        Compute  $\rho_i^{dh}(t_m - \hat{\tau}_{t_m})$  using (4.2):

$$\rho_i^{dh}(t_m - \hat{\tau}_{t_m}) = \rho_i^m(t_m)$$

        Compute  $h_{ex}(t_m - \hat{\tau}_{t_m})$  using (4.1)

        Compute  $n_0$  such that

$$t_m - \hat{\tau}_{t_m} = t_{m-1} - \hat{\tau}_{t_{m-1}} + n_0 t_s$$

**for**  $i = 1, \dots, n_0 - 1$  **do**

        Compute  $\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}} + it_s)$  (4.19):

$$\rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}} + it_s) = \frac{n_0 - i}{n_0} \rho_i^{dh}(t_{m-1} - \hat{\tau}_{t_{m-1}}) + \frac{i}{n_0} \rho_i^{dh}(t_m - \hat{\tau}_{t_m}) \quad (4.20)$$

        Compute  $h_{ex}(t_{m-1} - \hat{\tau}_{t_{m-1}} + it_s)$  using (4.1)

        Combine  $\hat{\mathbf{x}}(t_{m-1} - \hat{\tau}_{t_{m-1}} + (i - 1)t_s)$  with a nonlinear filter to compute the delayed estimate  $\hat{\mathbf{x}}(t_{m-1} - \hat{\tau}_{t_{m-1}} + it_s)$

**end for**

    Combine  $\hat{\mathbf{x}}(t_{m-1} - \hat{\tau}_{t_{m-1}} + (n_0 - 1)t_s)$  with a nonlinear filter to compute the delayed estimate  $\hat{\mathbf{x}}(t_m - \hat{\tau}_{t_m})$

**end if**

**end for**

---

## 4. SOLUTIONS TO THE DRAG-HEAD ESTIMATION PROBLEMS

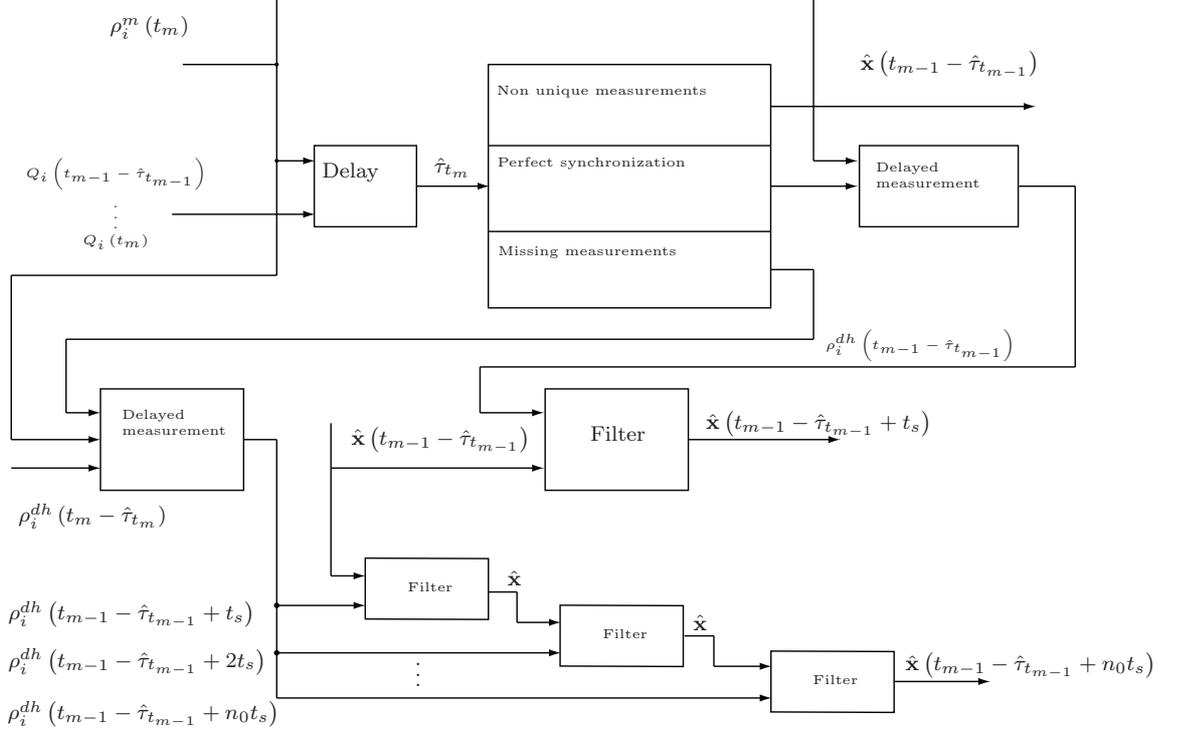


Figure 4.3: Iterative (delayed) state estimate. The algorithm starts with the estimate  $\hat{\mathbf{x}}(t_{m-1} - \hat{\tau}_{t_{m-1}})$  and computes the estimate  $\hat{\mathbf{x}}(t_m - \hat{\tau}_{t_m})$  and, if necessary, all the estimates in between.

- II. We consider the total mass  $m_t$  to be a known input to the filter rather than the state of the system.

The first assumption corresponds to  $k_{vh}$  being a dynamic uncertain parameter, the situation which has been discussed in Section 2.2. The second assumption is justified by the fact that the total mass of the ship  $m_t$  is measured on board of the ship with high accuracy and precision.

We show how this information is used by several nonlinear parametric filters discussed in Chapter 3. In particular we simulated the cutting process and to the generated data we applied EKF, UKF, CDF, and GSF. We compare the performance of these filters to determine which method is preferable as a solution to the Cutting Estimation Problem.

### 4.3.1 Derivation of the Discrete-Time Stochastic System

As was mentioned above, the objective of the discrete-time stochastic model of the cutting tool is to estimate the ratio  $k_{vh}$  between cutting forces and the horizontal cutting force coefficient  $k_{ch}$ .

### 4.3. Solutions to the Cutting Estimation Problem

---

Recall that by (2.15) the cutting process in continuous-time is defined by:

$$dv_{sh}(t) = \frac{1}{m_t(t)} (F_{th}(t) - F_{ch}(t)) dt + de_{vsh}, \quad (4.21a)$$

$$dk_{ch}(t) = 0dt + de_{ch}(t), \quad (4.21b)$$

$$dk_{vh}(t) = 0dt + de_{vh}(t), \quad (4.21c)$$

$$F_{vc} = \frac{x_2}{r_{vc}} k_{ch} h_c (\sin(\alpha_{lt} + \alpha_v) + k_{vh} \cos(\alpha_{lt} + \alpha_v)). \quad (4.21d)$$

In (4.21) the following variables are considered to be deterministic inputs to the system: the total mass  $m_t$ , the thrust force  $F_{th}$ , the cutting depth  $h_c$ , the dredging depth  $h_z$ , the visor angle  $\alpha_v$ , the angle of the lower suction pipe  $\alpha_{lt}$ , and the moment arm  $r_{vc}$ .

Note that in the Cutting Estimation Problem the measurements of  $h_c = h_{ex}$  are delayed. However, as was postulated in Section 4.2, we can consider the filtering problem independently of the delay problem. Thus, throughout this section we assume that the cutting depth  $h_c$  is known without delay. The final state estimate (with delayed measurements) is obtained by Algorithm 4.2.

Two variables are measured with uncertainty: the speed of the ship  $v_{sh}$  and the visor cylinder force  $F_{vc}$ , both assumed to be corrupted by the zero-mean, time-invariant Gaussian noises  $e_{vsh}^o$  and  $e_{F_{vc}}^o$  with standard deviations  $\sigma_{vsh}^o$  and  $\sigma_{F_{vc}}^o$ , respectively.

The discrete-time approximation of the velocity  $v_{sh}$  is obtained by the Euler discretization of (4.21a):

$$\frac{v_{sh,k} - v_{sh,k-1}}{T_s} = \frac{1}{m_{t,k-1}} (F_{th,k-1} - k_{ch,k-1} h_{c,k-1} (h_{z,k-1} + 10)) + e_{vsh}. \quad (4.22)$$

where  $k$  denotes the sampling step and  $e_{vsh}$  is a zero-mean Gaussian random variable with standard deviation  $\sigma_{vsh}$ .

The evolutions of the soil-dependent parameters  $k_{vh,k}$  and  $k_{ch}$  are described by discrete-time random-walk models:

$$k_{ch,k+1} = k_{ch,k} + T_s e_{ch}, \quad (4.23a)$$

$$k_{vh,k+1} = k_{vh,k} + T_s e_{vh}, \quad (4.23b)$$

where  $e_{vh}$  and  $e_{ch}$  are zero-mean Gaussian random variables with standard deviations  $\sigma_{vh}$  and  $\sigma_{ch}$ , respectively. These are approximations of the continuous-time models (4.21c)–(4.21b).

The dynamic model for the visor cylinder force  $F_{vc}$  is derived from (4.21d) in the following way. First, we note that the signal  $F_{vc}(t)$  is slowly varying with regard to the chosen sampling time  $T_s$ . In other words, we assume that with almost 100% probability the difference  $|F_{vc,k} - F_{vc,k-1}|$  is smaller than  $3\sigma_{vc}$  for all  $k$ . This means that the evolution of  $F_{vc}$  can be approximated by the following random walk:

$$F_{vc,k+1} = F_{vc,k} + T_s e_{vc}, \quad (4.24)$$

where  $e_{vc}$  is a zero-mean Gaussian random variable with standard deviation  $\sigma_{vc}$ . From (4.21d) and (4.24) we derive the final form of the discrete-time dynamical model for  $F_{vc}$ :

$$F_{vc,k+1} = \frac{x_2}{r_{vc,k}} k_{ch,k} h_{c,k} (h_{z,k} + 10) (\sin(\alpha_{lt,k} + \alpha_{v,k}) + \cos(\alpha_{lt,k} + \alpha_{v,k}) k_{vh,k}) + T_s e_{vc}. \quad (4.25)$$

## 4. SOLUTIONS TO THE DRAG-HEAD ESTIMATION PROBLEMS

---

The state, input, and output vectors of the system are defined by:

$$\mathbf{u} = \begin{pmatrix} F_{th} \\ h_c(h_z + 10) \\ \alpha_{lt} + \alpha_v \\ \frac{x_2}{r_{vc}} \\ m_t \end{pmatrix}, \quad \mathbf{x} = \begin{pmatrix} v_{sh} \\ k_{ch} \\ k_{vh} \\ F_{vc} \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} v_{sh} \\ F_{vc} \end{pmatrix}.$$

Then, the final form of the state-space model is expressed as:

$$x_{1,k+1} = x_{1,k} + \frac{T_s}{u_{5,k}} (u_{1,k} - x_{2,k}u_{2,k}) + e_{vsh} \quad (4.26a)$$

$$x_{2,k+1} = x_{2,k} + e_{ch}, \quad (4.26b)$$

$$x_{3,k+1} = x_{3,k} + e_{vh}, \quad (4.26c)$$

$$x_{4,k+1} = u_{4,k}u_{2,k}x_{2,k} (\sin(u_{3,k}) + \cos(u_{3,k})x_{3,k}) + e_{vc}, \quad (4.26d)$$

$$y_{1,k} = x_{1,k} + e_{vsh}^o, \quad (4.26e)$$

$$y_{2,k} = x_{4,k} + e_{Fvc}^o, \quad (4.26f)$$

### 4.3.2 Numerical Simulations

#### Setting

We simulated the cutting process during which the soil-dependent parameters are changing. In the real process this corresponds to sailing through a dredging area with different types of in situ material. At first, the soil-dependent parameters are approximately equal to  $k_{ch} = 8$ , and  $k_{vh} = 0.15$ . After 13 seconds the cutting tool encounters a different type of soil, which is characterized by parameters values close to  $k_{ch} = 12.5$ , and  $k_{vh} = 0.18$ . Finally, after 26 seconds the soil characteristics change to approximately  $k_{ch} = 11$ , and  $k_{vh} = 0.16$ . The sampling time is set to  $T_s = 0.05$ [s]. To better illustrate the tracking properties of the filtering algorithms employed, each change in  $k_{ch}$  and  $k_{vh}$  is set to be step-like, which models dramatic changes in the soil-type of the dredged material.

We start the simulation from the initial state

$$\mathbf{x}_0 = (1 \quad 8 \quad 0.15 \quad 7.2 \cdot 10^4)^T$$

The process and observation noises are assumed to be zero-mean Gaussian variables with the parameters reported in Table 4.1.

Table 4.1: Characteristics of the noises used in the simulation

Variable	$\sigma_{vsh}$	$\sigma_{ch}$	$\sigma_{vh}$	$\sigma_{vc}$	$\sigma_{vsh}^o$	$\sigma_{Fvc}^o$
Value	0.001	10	10	100	0.1	$10^4$
Unit	[m/s]	[kg/s <sup>2</sup> ]	N.A.	[N]	[m/s]	[N]

The EKF, UKF, CDF, and GSF described in Chapter 3 are applied to the simulated data, obtained with the discrete-time cutting model described in Section 4.3.1. First, each filter application is discussed separately as each of them has different parameters to be tuned. Then the results obtained by all filters are compared to determine which method provides the best solution to the Cutting Estimation Problem. To allow for a fair comparison, all the filters under investigation are tuned with the same process and measurements noises. Namely, the filters'

### 4.3. Solutions to the Cutting Estimation Problem

noises are equal to the “true” (simulated) noises reported in Table 4.1, except for the noises of  $k_{ch}$  and  $k_{vh}$ . For the tested filters, the standard deviations of the variables  $e_{ch}$  and  $e_{vh}$  are set to  $20\sigma_{ch}$  and  $20\sigma_{vh}$ , respectively. The use of higher variances in filters design resembles the situation where there is a high model uncertainty.

As we are interested in the estimation of the soil-dependent parameters  $k_{ch}$  and  $k_{vh}$ , in what follows we show only the outcomes of the estimation of the states  $x_{2,k}$  and  $x_{3,k}$ . The initial offset of the EKF, the UKF, and the CDF has been set to 130% of the true initial state of the process, i.e., the initial state for the filters is given by:

$$\mathbf{x}_0 = \left( 1.3 \quad 10.4 \quad 0.195 \quad 9.36 \cdot 10^4 \right)^T,$$

and the initial covariance of the system is given by:

$$P_0 = \begin{bmatrix} 10^{-4} & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}.$$

The initial offset of the GSF is discussed separately. This is because the initial state of the GSF is computed as an average of initial states of GS-terms used by the filter, each having different offset.

#### Results Obtained by the Extended Kalman Filter

Let us now apply the EKF to the system (4.26) described in the previous section. The estimates obtained by the EKF are presented in Figure 4.4. We can see that the estimates are very accurate in tracking the true states of the system. In the case of  $k_{ch}$  there is no overshoot, whereas when estimating the state  $k_{vh}$  the overshoot does not exceed 32% of the nominal value. The settling times are comparable for both signals and are in the range of 2 – 3 seconds. These results suggest that the EKF gives a satisfactory solution to the Cutting Estimation Problem. Further analysis of this example is carried out in Section 4.3.3.

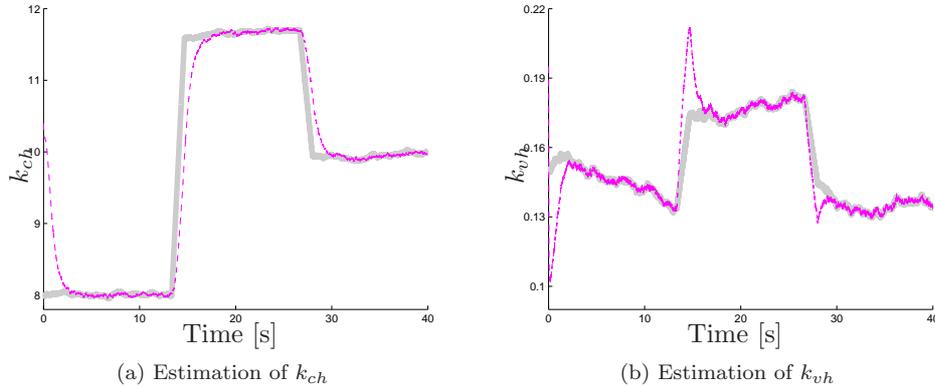


Figure 4.4: The EKF applied to the system (4.26). The thick solid line represents the true state of the system, the dashed thin line denotes the estimates of  $k_{ch}$  (left) and  $k_{vh}$  (right).

## 4. SOLUTIONS TO THE DRAG-HEAD ESTIMATION PROBLEMS

### Results Obtained by the Unscented Kalman Filter

Figure 4.5 shows the estimates of the horizontal cutting coefficient  $k_{ch}$  and the ratio  $k_{vh}$  between the cutting forces obtained by the UKF applied to the system (4.26). It can be seen that the performance of the UKF is comparable to the performance of the EKF. Namely, there is no overshoot when tracking the variable  $k_{ch}$  and there is less than 32% overshoot when tracking the variable  $k_{vh}$ . Similarly, the settling times are between 2 and 3 seconds. The detailed comparison carried out in Section 4.3.3 reveals that for system defined by (4.26) the UKF-produced estimates are slightly more accurate than the estimates produced by the EKF.

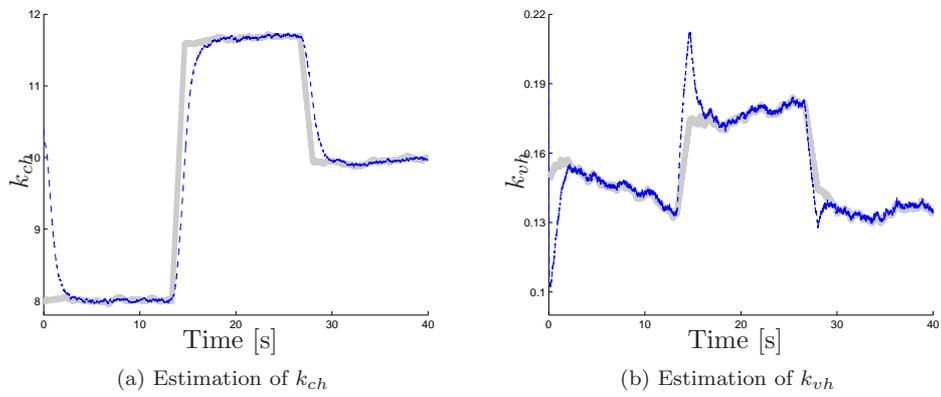


Figure 4.5: The UKF applied to the system (4.26). The thick solid line represents the true state of the system, the dashed thin line denotes the estimates of  $k_{ch}$  (left) and  $k_{vh}$  (right).

### Results Obtained by the Central Difference Filter

We applied the CDF to the simulated data for three different settings of the filter's parameter  $h$ :  $h = 0.001$ ,  $h = 1$ ,  $h = 10^4$ . As can be seen from Figure 4.6 there is no significant difference between the estimates of the state  $k_{ch}$ . However, the choice of the step size  $h$  influences the filter's performance when estimating the state  $k_{vh}$ . The CDF with  $h = 1$  has faster response time to the changes in the simulated  $k_{vh}$  than the remaining filters. Furthermore, in steady state, the CDF-produced estimates of  $k_{vh}$  are more accurate for  $h = 1$  than for other values of  $h$ . On the other hand, after the system is excited by a rapid change in  $k_{vh}$ , a faster response of the CDF with  $h = 1$  results in a larger overshoot (31.8%) when compared with the two other CDFs (29.9% for  $h = 0.001$ , and 24% for  $h = 10^4$ ).

### Results Obtained by the Gaussian Sum Filter

We applied the GSF as a collection of parallel UKFs to the system defined by (4.26) in three different settings. The filters differ by the number of terms in the GS that approximates the posterior distribution ( $K \in \{2, 3, 4\}$ ). Both the process and the observation noises are additive and Gaussian, thus each of them is represented by a single Gaussian. Therefore, as was mentioned in Section 3.3.3, the number of terms in the GS is fixed, and there is no need for employing the reduction techniques described in Section 3.3.3.

As it was explained in Section 3.3.3 with a larger number of GS-terms it is easier to model the non-Gaussian uncertainty in the state. This also applies when it comes to handling the

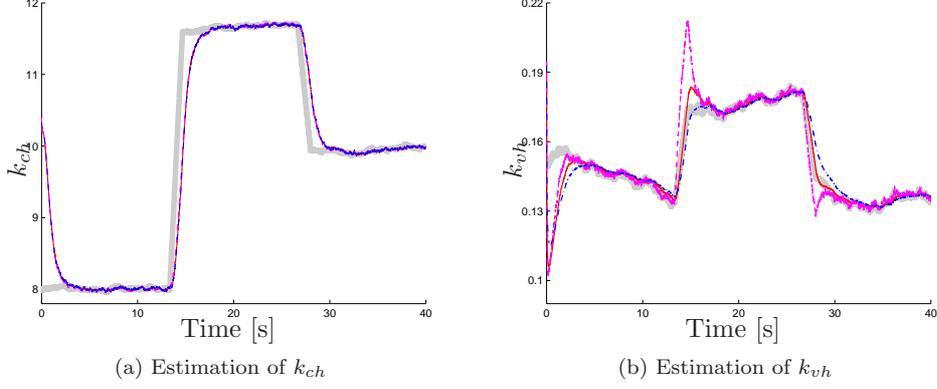


Figure 4.6: The CDF applied to the system (4.26). The thick solid line represents the true state of the system, the thin lines denote the estimates. The thin solid line represents the CDF with  $h = 0.001$ , the thin dashed line represents the CDF with  $h = 1$ , and the thin dotted-dashed line represents the CDF with  $h = 10^4$ .

uncertainty in the initial state. Namely, with the GSF we have the ability to select several independent, equally weighted, GS terms with equal covariances to represent the uncertainty in the initial state of the system. In this way we can cover the uncertainty region with higher precision (see Figure 3.6). Because of that a faster convergence to the true state of the system can be achieved.

This is confirmed by the results obtained for the system (4.26) with the use of the GSFs starting from:

I. the two-term GSF:

$$\mu_1 = \begin{pmatrix} 1.3 \\ 10.4 \\ 0.195 \\ 9.36 \cdot 10^4 \end{pmatrix} \quad \mu_2 = \begin{pmatrix} 1.3 \\ 5.6 \\ 0.105 \\ 9.36 \cdot 10^4 \end{pmatrix}$$

II. the three-term GSF:

$$\mu_1 = \begin{pmatrix} 1.3 \\ 10.4 \\ 0.195 \\ 9.36 \cdot 10^4 \end{pmatrix} \quad \mu_2 = \begin{pmatrix} 1.3 \\ 5.6 \\ 0.105 \\ 9.36 \cdot 10^4 \end{pmatrix} \quad \mu_3 = \begin{pmatrix} 1.3 \\ 10.4 \\ 0.105 \\ 9.36 \cdot 10^4 \end{pmatrix}$$

III. the four-term GSF:

$$\mu_1 = \begin{pmatrix} 1.3 \\ 10.4 \\ 0.195 \\ 9.36 \cdot 10^4 \end{pmatrix} \quad \mu_2 = \begin{pmatrix} 1.3 \\ 5.6 \\ 0.105 \\ 9.36 \cdot 10^4 \end{pmatrix} \quad \mu_3 = \begin{pmatrix} 1.3 \\ 10.4 \\ 0.105 \\ 9.36 \cdot 10^4 \end{pmatrix} \quad \mu_4 = \begin{pmatrix} 1.3 \\ 5.6 \\ 0.195 \\ 9.36 \cdot 10^4 \end{pmatrix}$$

## 4. SOLUTIONS TO THE DRAG-HEAD ESTIMATION PROBLEMS

In all three cases the initial covariance for each GS-term is equal and is given by:

$$P_0 = \begin{bmatrix} 10^{-4} & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}.$$

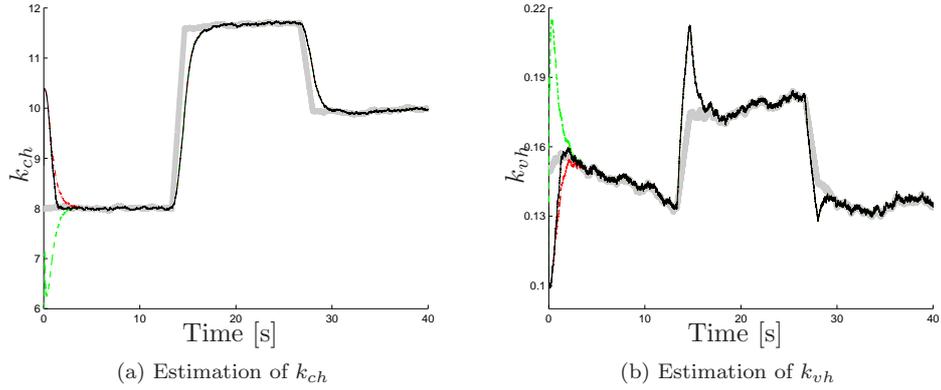


Figure 4.7: The GSF applied to the system (4.26). The thick solid line represents the true state of the system, the thin lines denote the estimates. The thin solid line represents the GSF with four GS terms, the thin dotted-dashed line represents the GSF with three GS terms, and the thin dashed line represents the GSF with two GS terms. The initial conditions for the filters are computed as the averages of two, three, or four GS-terms.

As can be observed in Figure 4.7, at the beginning of the estimation the four-term GSF converges faster to the true state of the process than the remaining GSFs. After a short time the weights associated with the GS terms degenerate, which means that the equilibrium state is reached: most of the mass is focused on few terms and the remaining terms have negligible weight [Kotecha and Djurić, 2003b]. In our example the number of significant terms is: 1 for the two-term GSF, 1 for the three-term GSF, and 2 for the four-term GSF. That means that after they reach a steady state, the two-term GSF and the three-term GSF act just as a UKF. The four-term GSF in its steady state acts as the convex combination of two UKFs. The degeneration phenomenon is advantageous from the computational perspective because it allows to disregard the terms with insignificant weights, and hence to propagate only a reduced number of GS terms. On the other hand, the diversity that boosted the convergence at the beginning of the estimation is lost. To avoid the degeneration phenomenon one can employ a resampling procedure [Kotecha and Djurić, 2003b].

### 4.3.3 Discussion

Let us now compare the results obtained by the filters presented in previous sections.

#### Performance Comparison

We define the convergence time as the minimal time  $t_c$ , such that for times  $t > t_c$  the estimate stays within the 1%-neighborhood of the true state of the signal. The overshoot is defined as the maximum difference between the filtered and the true signal.

### 4.3. Solutions to the Cutting Estimation Problem

Due to the rapid changes in the true signal it might happen that the estimate diverges from the true value of the signal. In such a situation it is reasonable to recompute the convergence time and the overshoot. In our simulations we repeat these computations three times: at the beginning of the estimation, at the growth at 13-th second, and the decrease at 26-th second.

Table 4.2: Characteristics of the EKF, the CDF, the UKF, and the GSF used to estimate  $k_{vh}$ . The convergence times and the overshoots are computed at three times: at the beginning of the simulation, at 13-th second of the simulation, and at 26-th second of the simulation.

Filter	Convergence time (seconds)			Overshoot (%)			MSE
EKF	2.19	3.25	2.75	31.9	22.9	11.7	8.5
UKF	2.1	3.26	2.66	31.8	23.2	11.5	8.54
CDF $h = 10^{-3}$	2.7	3.26	1.48	30	6	3.7	5.2
CDF $h = 1$	2.1	3.26	2.66	31.8	23.2	11.5	8.56
CDF $h = 10^4$	3.53	1.84	5.16	24.7	10.1	10.1	5.42
GSF $GS = 2$	2.87	3.26	2.66	42.4	23.2	11.5	10.95
GSF $GS = 3$	2.14	3.26	2.66	34	23.2	11.5	9.35
GSF $GS = 4$	1.36	3.26	2.66	34	23.2	11.5	8.67

In Table 4.2 the EKF, UKF, CDFs, and GSFs are compared by means of three quantitative measures: the convergence time, the overshoot, and the Mean Square Error (MSE). Table 4.2 presents only the results of the estimation of  $k_{vh}$ . The analysis of the estimation results of  $k_{ch}$  are omitted because in that case all of the discussed filters achieve similar performance. Indeed, all the estimated signals look the same, except for the four-terms GSF which converges faster to the true  $k_{ch}$  in the beginning.

#### Recommendations

The simulation results suggest that each of the discussed nonlinear filters gives an adequate solution to the Cutting Estimation Problem. The detailed analysis of the quantitative measures presented in Table 4.2 suggests that the EKF, the UKF, the CDF with parameter  $h = 1$ , and the two and three-term GSFs are virtually indistinguishable performance-wise.

The CDF with parameters  $h = 10^{-1}$ , and  $h = 10^4$  have the smallest MSE errors, and the smallest overshoots. The four-term GSF has the fastest convergence time during the first filtering phase (similarly to the case of filtering the state  $k_{si}$ ). In the second and third phase the shortest settling times have been achieved by the CDF with  $h = 10^4$  and the CDF with  $h = 10^{-3}$ , respectively.

To conclude, when the uncertainty in the initial states of the system is large the four-term GSF converges the fastest to the true value of the signal. If the uncertainty in the initial states is small, e.g., we know the exact position of the system, to minimize the errors and to preserve fast convergence times it is recommended to use one of the variations of the CDF (with  $h = 10^{-3}$  or  $h = 10^4$ ).

## 4.4 Solutions to the Cutting and Jetting Estimation Problem

In this section we give solutions to the Cutting and Jetting Estimation Problem defined in Chapter 2. We start by briefly recalling the system under the consideration.

### System Specifications

The main objective of the Cutting and Jetting Estimation is to estimate two soil-dependent parameters: the cutting force coefficient  $k_{ch}$  and in situ permeability  $k_{si}$ . These parameters are unknown and can change dynamically during the dredging operation when the type of the excavated material changes.

Throughout this section we make the following assumptions:

- I. The ratio  $k_{vh}$  is a known constant parameter, equal for all the soil types.
- II. The following variables are known without a measurement error: the geometrical parameters of the Drag-Head  $C_{dh}, x_2$ ; the jet nozzle pressure  $p_j$ , the jet water flow  $Q_{w,j}$ , the speed of the ship  $v_{sh}$ , the moment arm  $r_{vc}$ , the visor cylinder force  $F_{vc}$ , the angle of the lower suction pipe  $\alpha_{lt}$ , the visor angle  $\alpha_v$ , and the dredging depth  $h_z$ .
- III. The excavation depth  $h_{ex}$  is measured with an error.

Then, the cutting input  $u_c$  and the jetting input  $u_j$  are defined by:

$$u_c := \frac{r_{vc}}{x_2} F_{vc} (\sin(\alpha_{lt} + \alpha_v) + k_{vh} \cos(\alpha_{lt} + \alpha_v))^{-1} (h_z + 10)^{-1}, \quad (4.27a)$$

$$u_j := C_{dh} p_j^{0.5} Q_{w,j} v_{sh}^{-1}, \quad (4.27b)$$

By (2.10) and (2.11) and the fact that  $h_{ex} = h_j + h_c$  (see (2.19)) we get:

$$h_{ex} = u_c k_{ch}^{-1} + u_j k_{si}^p, \quad (4.28)$$

Note that apart from soil-dependent parameters  $k_{ch}$  and  $k_{si}$  all the variables used in the models for  $u_c$  and  $u_j$  are assumed to be known. Furthermore, we also assume that the excavation depth  $h_{ex}$ , which is equal to the visor depth  $h_v$ , is measured. To summarize, (4.28) describes a dynamic model with two known input variables, one measured variable, and two dynamic parameters that need to be estimated online.

### Correlation Between Soil-Dependent Parameters

Because both the horizontal cutting force coefficient  $k_{ch}$  and the in situ permeability  $k_{si}$  depend on the type of soil that is excavated by the Drag-Head, the parameters are correlated. It is difficult to establish accurate relations between these parameters and the in situ soil. Nevertheless, empirical studies suggest the following approximate values for the two parameters for eight common types of soil:

#### 4.4.1 Formulation of the Discrete-Time Stochastic System

First, let us assume that the measured variable  $h_{ex}$  is corrupted by a zero-mean Gaussian noise  $e_{hex}^o$  with a constant standard deviation  $\sigma_{hex}^o$ . Then, for every sampling step  $k$ , (4.28) is given by:

$$h_{ex,k} = u_{c,k} k_{ch,k}^{-1} + u_{j,k} k_{si,k}^p + e_{hex}^o. \quad (4.29)$$

#### 4.4. Solutions to the Cutting and Jetting Estimation Problem

Table 4.3: Values of  $k_{ch}$  and  $k_{si}$  depending on the soil type.

Soil type	Soil type A (Fine)		Soil type B (Medium)		Soil type C (Medium)		Soil type D (Coarse)	
	Medium	Dense	Medium	Dense	Medium	Dense	Medium	Dense
SPT	20	40	20	40	20	40	20	40
$d_m [mm]$	0.10	0.10	0.24	0.24	0.45	0.45	1.30	1.30
$\rho_s [kg/m^3]$	2030	2140	2010	2120	2030	2140	2040	2150
$k_{ch}$	$9.87 \cdot 10^4$	$1.16 \cdot 10^5$	$9.56 \cdot 10^4$	$1.12 \cdot 10^5$	$9.53 \cdot 10^4$	$1.11 \cdot 10^5$	$8.89 \cdot 10^4$	$1.03 \cdot 10^5$
$k_{si} [m/s]$	$3.59 \cdot 10^{-5}$	$1.6 \cdot 10^{-5}$	$2.75 \cdot 10^{-4}$	$1.3 \cdot 10^{-4}$	$4.06 \cdot 10^{-4}$	$1.81 \cdot 10^{-4}$	$2.86 \cdot 10^{-3}$	$1.23 \cdot 10^{-3}$

Note that the changes in the in situ soil are slow relative to the sampling speed of the measured variables. Therefore, we can assume that the time-varying parameters  $k_{ch}$  and  $k_{si}$  are piecewise constant over a given time interval. To be precise we assume that the soil-dependent parameters  $k_{ch}$  and  $k_{si}$  can change their value every 1[s], during which we collect  $N$  measurements of the variables  $h_{ex}$ ,  $u_c$ , and  $u_j$ . In other words, if  $k$  is a time index such that the parameters  $k_{ch}$  and  $k_{si}$  are constant over the interval  $[k, k+1)$  [s] the measurements are collected at time instances  $(k + l/N)$  [s] for  $l = 0, \dots, N-1$ . Then, the system equations are written as:

$$h_{ex, k+l/N} = u_{c, k+l/N} k_{ch, k}^{-1} + u_{j, k+l/N} k_{si, k}^p + e_{h_{ex}}^o. \quad (4.30)$$

Note that the minimal number of measurements per time step that makes the system solvable is  $N = 2$ . Thus, a dynamic filter based on model (4.30) at each time step  $k = 1, 2, \dots$  processes the  $N \times 3$ -measurement matrix

$$\begin{bmatrix} u_{c, k} & u_{j, k} & h_{ex, k} \\ \vdots & \vdots & \vdots \\ u_{c, k + \frac{N-1}{N}} & u_{j, k + \frac{N-1}{N}} & h_{ex, k + \frac{N-1}{N}} \end{bmatrix}.$$

Next, let us specify the discrete-time model for the parameters  $k_{ch}$  and  $k_{si}$ . We investigate two approaches, each leading to a different dynamical system:

- I. The parameters  $k_{ch}$  and  $k_{si}$  are uncorrelated,
- II. The parameters  $k_{ch}$  and  $k_{si}$  are correlated.

We discuss the two cases separately.

##### Particle Filter for Uncorrelated $k_{ch}$ and $k_{si}$

Let us now discuss the first approach. The filter has two one-dimensional states that correspond to the parameters  $k_{ch}$  and  $k_{si}$ , two  $N$ -dimensional known inputs that correspond to the values of the cutting input  $u_c$  and the jetting input  $u_j$  recorded during the interval  $[kT_s, (k+1)T_s)$ , and one  $N$ -dimensional observation vector that corresponds to the measurements of excavation depth  $h_{ex}$ , defined in (4.30), taken during the interval  $[kT_s, (k+1)T_s)$ .

The state, input and output vectors of the system are defined by:

$$\mathbf{u} = \begin{pmatrix} u_c \\ u_j \end{pmatrix}, \quad \mathbf{x} = \begin{pmatrix} k_{ch} \\ k_{si} \end{pmatrix}, \quad \mathbf{y} = h_{ex}.$$

Assuming that  $k_{ch}$  and  $k_{si}$  are uncorrelated states of the system we model the two parameters as uncorrelated random walks with noises  $e_{k_{ch}}, e_{k_{si}}$  with standard deviations  $\sigma_{k_{ch}}$  and  $\sigma_{k_{si}}$ ,

## 4. SOLUTIONS TO THE DRAG-HEAD ESTIMATION PROBLEMS

respectively. Thus, the dynamical model of the filter is given by:

$$x_{1,k+1} = x_{1,k} + e_{kch}, \quad (4.31a)$$

$$x_{2,k+1} = x_{2,k} + e_{k_{si}}, \quad (4.31b)$$

$$y_k = [u_{1,k} \ u_{2,k}] \begin{bmatrix} x_{1,k}^{-1} & x_{2,k}^p \end{bmatrix}^T + E_{hex}^o, \quad (4.31c)$$

where, for  $i = 1, 2$

$$u_{i,k} = [u_{i,k}, u_{i,k+1/N}, \dots, u_{i,k+(N-1)/N}]^T, \quad (4.32)$$

and  $E_{hex}^o$  is a zero-mean  $N$ -dimensional Gaussian with standard deviation  $\Sigma_{hex}^o$  defined by

$$\Sigma_{hex}^o = \begin{bmatrix} \sigma_{hex}^o & & 0 \\ & \ddots & \\ 0 & & \sigma_{hex}^o \end{bmatrix}. \quad (4.33)$$

### Particle Filter for Correlated $k_{ch}$ and $k_{si}$

Now, let us investigate the second approach. Since both parameters  $k_{ch}$  and  $k_{si}$  describe the physical properties of the in situ soil, intuitively, they should be correlated. To determine this correlation we will utilize the information from Table 4.3. Namely, we describe the parameter  $k_{ch}$  as a function of the parameter  $k_{si}$  using the values specified in Table 4.3. The numbers reported in the aforementioned table clearly suggest that the relation between  $k_{si}$  and  $k_{ch}$  is not linear. Therefore, we decided to fit a polynomial to the 8 points given in Table 4.3. Figure 4.8 shows two approximations with polynomials of second and third order.

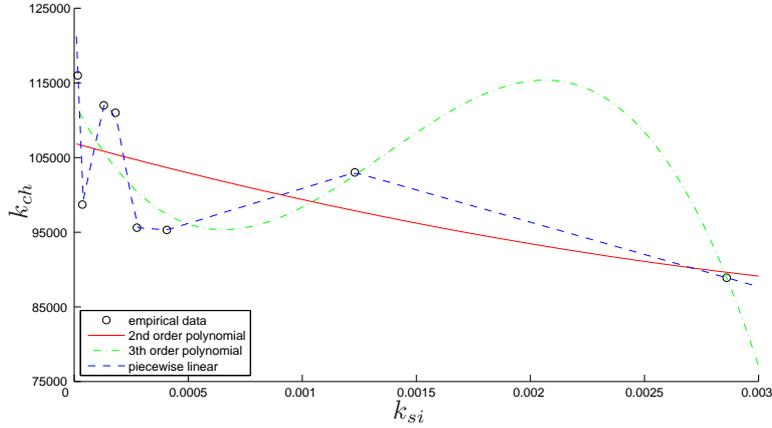


Figure 4.8: Function fits to the 8 data points (circles) given in Table 4.3. The solid line represents the 2nd order polynomial fit, the dotted line represents the 3rd order polynomial fit and the dashed line corresponds to the piecewise linear fit.

Since the 2nd order polynomial does not resemble the relation between data we will use only the 3rd order polynomial, which matches the data much better for  $k_{si} < 0.0015$ , which is the most common situation during the actual dredging. This polynomial is given by:

$$f_p(x) = 11.23 \cdot 10^4 - (5803 \cdot 10^4) x + (584 \cdot 10^8) x^2 + (1433 \cdot 10^{10}) x^3. \quad (4.34)$$

#### 4.4. Solutions to the Cutting and Jetting Estimation Problem

---

Alternatively, one could also use a simple linear interpolation between the data points as an approximate functional relation between  $k_{si}$  and  $k_{ch}$ . This can also be seen in Figure 4.8.

$$f_a(x) = \begin{cases} -(8693 \cdot 10^5) \cdot x + 129910 & \text{for } 0 < x < 3.59 \cdot 10^{-5} \\ (1413 \cdot 10^5) \cdot x + 93626 & \text{for } 3.59 \cdot 10^{-5} \leq x < 13 \cdot 10^{-5} \\ -(196 \cdot 10^5) \cdot x + 114549 & \text{for } 13 \cdot 10^{-5} \leq x < 18.1 \cdot 10^{-5} \\ -(1638 \cdot 10^5) \cdot x + 140653 & \text{for } 18.1 \cdot 10^{-5} \leq x < 27.5 \cdot 10^{-5} \\ -(23 \cdot 10^5) \cdot x + 96230 & \text{for } 27.5 \cdot 10^{-5} \leq x < 40.6 \cdot 10^{-5} \\ (93 \cdot 10^5) \cdot x + 91506 & \text{for } 40.6 \cdot 10^{-5} \leq x < 123 \cdot 10^{-5} \\ -(87 \cdot 10^5) \cdot x + 113640 & \text{for } 123 \cdot 10^{-5} \leq x \end{cases} \quad (4.35)$$

Then, using the same notation as in (4.31), the dynamical system for the correlated  $k_{ch}$  and  $k_{si}$  is given by:

$$x_{1,k+1} = f_{p,a}(x_{2,k+1}) + e_{kch}, \quad (4.36a)$$

$$x_{2,k+1} = x_{2,k} + e_{k_{si}}, \quad (4.36b)$$

$$y_k = [u_{1,k} \ u_{2,k}] \begin{bmatrix} x_{1,k}^{-1} & x_{2,k}^p \end{bmatrix}^T + E_{hex}^o \quad (4.36c)$$

where  $f_{p,a}$  is either the polynomial  $f_p$  defined in (4.34) or the piecewise linear function  $f_a$  defined by (4.35).

#### Cascaded Filter for Correlated $k_{ch}$ and $k_{si}$

An alternative method of using the knowledge regarding the correlations between the parameters  $k_{ch}$  and  $k_{si}$  given in Table 4.3 is to use a cascaded filter.

We have developed such a cascaded filter and it acts in the following manner:

- I. First, the PF based on the system (4.31) produces the estimates  $\hat{k}_{si}$ .
- II. Next, the estimate  $\hat{k}_{si}$  is fed to the Steady State Identification (SSI) filter [Bhat and Saraf, 2004].
- III. After the steady state has been detected by the SSI filter, the value of  $\hat{k}_{si}^s$  is fed to the Bayesian Filter (BF) as a prior information.
- IV. Finally, the BF produces the estimate  $\hat{k}_{ch}$  of the parameter  $k_{ch}$ .

The schematic view of this algorithm is presented in Figure 4.9.

The BF from step IV acts in three steps:

- I. First, the estimate  $\hat{k}_{si}^s$  of the parameter  $k_{si}$  and the knowledge of the relation between  $k_{si}$  and  $k_{ch}$  are used to determine a probable value of  $k_{ch}$ . This value cannot be determined with a complete certainty due to the uncertain character of the estimate  $\hat{k}_{si}^s$  and the approximate knowledge of the relations between  $k_{si}$  and  $k_{ch}$ . Therefore, the probable value of  $k_{ch}$  is a random variable and the corresponding probability distribution  $\pi_0$  is called the prior distribution.
- II. In the second step, the prior  $\pi_0$  from the previous step is combined with the inputs and the measurements denoted by  $U$  and  $Y$  to obtain the posterior distribution  $\pi_1$  of the parameter in question  $k_{ch}$ . The posterior  $\pi_1$  is derived from Bayes theorem:

$$\pi_1(k_{ch}) = \frac{\mathbb{P}(Y|k_{ch}, U) \pi_0(k_{ch})}{\int \mathbb{P}(Y|k_{ch}, U) \pi_0(k_{ch}) dk_{ch}}, \quad (4.37)$$

## 4. SOLUTIONS TO THE DRAG-HEAD ESTIMATION PROBLEMS

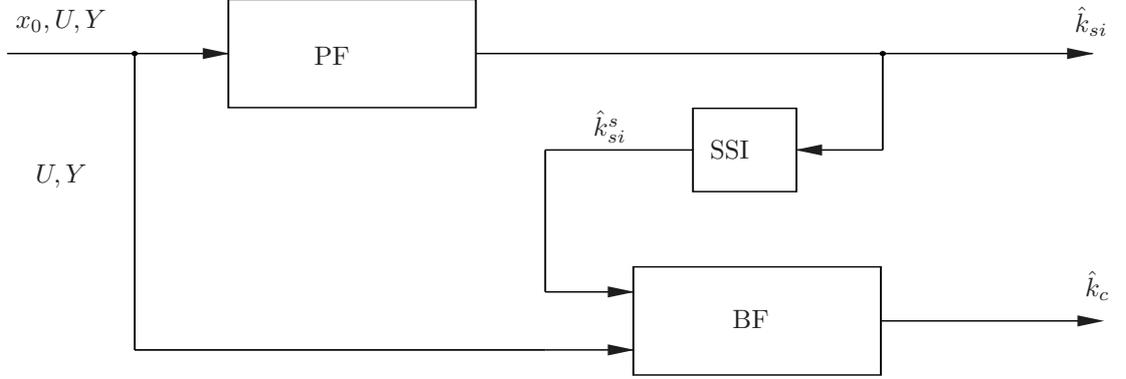


Figure 4.9: Block diagram of the cascaded filter.

where  $\mathbb{P}(Y|k_{ch}, U)$  is a PDF of the observation (likelihood), derived from (4.31c) under the assumption that  $k_{si} = \hat{k}_{si}^s$ .

- III. Finally, the Bayes estimate  $\hat{k}_c$  is computed as the expectation of the variable  $k_{ch}$  with respect to measure  $\pi_1$

$$\hat{k}_c = \int k_{ch} \pi_1(k_{ch}) dk_{ch}, \quad (4.38)$$

which is the optimal estimate in the Mean Squared Error (MSE) sense [Ross, 2009].

Note that the integral (4.38) cannot be solved analytically. Thus, to obtain the estimate we used Monte Carlo approximation of the expected value (4.38). This should not be confused with the way in which the PF produces the estimate of  $k_{si}$ . In the PF, at the end of each filtering step the full approximation of the posterior distribution is saved and transmitted to the next step as a new prior. In the BF that we have described above there is no memory in the system. Indeed, the prior is fully determined by  $\hat{k}_{si}^s$ , which comes as an input to the BF block of the cascaded filter.

The use of the SSI filter in step II allows the use of the BF in the consecutive step without the necessity of an extensive tuning procedure. Without the SSI, the BF is sensitive to the choice of the prior distribution  $\pi_0$ . As has been observed in numerical simulations, an incorrect choice of the prior results in large oscillations of the estimates of the  $k_{ch}$ . Application of a fast and simple SSI allows to circumvent this undesirable behavior. On the downside, as we freeze the estimates of the  $k_{ch}$  during the transient phase, we cannot rely on the estimates of the  $k_{ch}$  before the steady state is reached.

### 4.4.2 Numerical Simulations

Let us see how the methods presented in the previous section work when applied to data obtained from numerical simulations of the excavation process.

#### Setting

We have simulated the excavation process during which the in situ soil changes dynamically. The simulation is divided into five equal time intervals, each characterized by a different in situ soil type according to Table 4.3. We simulate the excavation process (4.27)–(4.28) over 160 time steps, where each time step corresponds to 1 [s]. Furthermore, we assume that the measurements

#### 4.4. Solutions to the Cutting and Jetting Estimation Problem

are taken with a frequency of 10 [Hz]. At the beginning of the process the excavated soil is coarse (Soil type D with medium packing). Then, after 32 [s] the Drag-Head encounters densely packed fine soil (Soil type A with dense packing). After another 32 [s] the excavated soil becomes finer (Soil type C with medium packing) and then, again after 32 [s], the soil becomes fine (Soil type A with medium packing). Finally, for the last 32 [s] of the simulation the Drag-Head slides over densely packed medium soil (Soil type B with dense packing). For each of these soil types the corresponding parameters  $k_{ch}$  and  $k_{si}$  have been defined according to the Table 4.3.

The input variables  $u_c$  and  $u_j$  are set to 16 operating points that are repeated periodically over the whole simulation horizon independently of changes in soil types. The operating points have been chosen to resemble the states of the Drag-Head system that are commonly met during dredging operations (see Tables A.1 and A.2). Because the time scope of the simulation is 160[s] and we assume 10 measurements are collected per second, the cutting input  $u_c$  and the jetting input  $u_j$  are periodic signals (with the period equal to 1.6[s]). Hence the periods of  $u_c$  and  $u_j$  do not coincide with the update step of the parameters. This results in a quasi-periodic character of the simulated signals. The standard deviation of the measurement noise  $e_{hex}^o$  is set to  $\sigma_{hex}^o = 0.03$ .

We have simulated two excavation scenarios:

- A The regime where the jetting depth  $h_j$  dominates the cutting depth  $h_c$ , which contributes very weakly to the total excavation depth  $h_{ex}$ . The simulation results of this scenario are presented in Figure 4.10a.
- B The regime where the jetting depth  $h_j$  and the cutting depth  $h_c$  contribute comparably to the total excavation depth  $h_{ex}$ . The simulation results of this scenario are presented in Figure 4.10b.

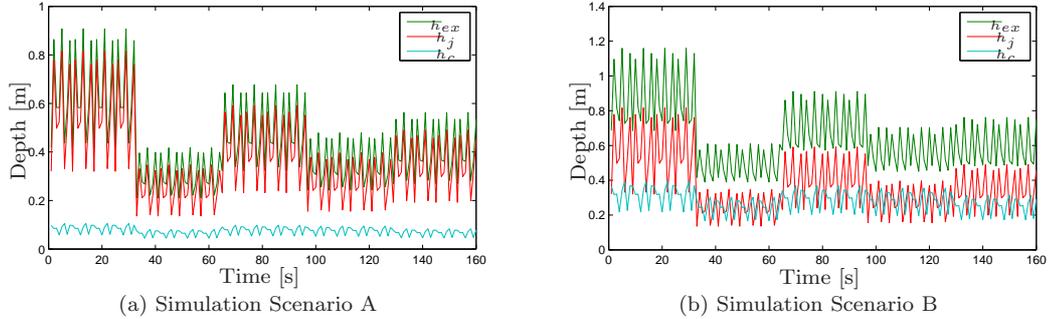


Figure 4.10: Simulation of the excavation process for dynamically changing parameters. The excavation depth  $h_{ex}$  is a sum of the cutting depth  $h_c$  and the jetting depth  $h_j$ , Scenario A (left) and Scenario B(right).

#### Results Obtained by the Particle Filter for Uncorrelated $k_{ch}$ and $k_{si}$

We have applied the PF based on system (4.31) to the simulated signals presented in Figure 4.10a. To make the method feasible for online application with the sampling time equal to 1[s] we use the PF with 1000 particles. The resampling threshold is set to  $N_T = 1000$ , which means that the resampling occurs each iteration. The estimates obtained by the PF can be seen in Figure 4.11.

#### 4. SOLUTIONS TO THE DRAG-HEAD ESTIMATION PROBLEMS

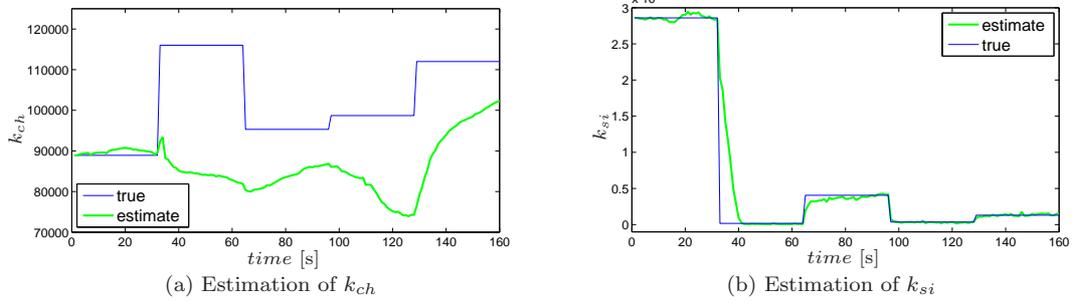


Figure 4.11: Estimation results obtained by the PF using 1000 particles with resampling (resampling threshold  $N_T = 1000$ ) applied to the data presented in Figure 4.10a. The thick line shows the average of ten independent runs of the PF.

From Figure 4.11 it can be observed that the algorithm is able to detect the moments when the excavated soil changes and it can quite accurately estimate the value of  $k_{si}$ . However, the estimation of  $k_{ch}$  is very inaccurate which makes the filter useless from the practical point of view.

One could argue that poor estimate of  $k_{ch}$  is due to the fact that the cutting height  $h_c$  very weakly contributes to the total excavation depth  $h_{ex}$  as can be observed in Figure 4.10a. To investigate this possibility we have applied the same PF that we used in the previous simulation (based on system (4.31), using 1000 particles and with the resampling threshold  $N_T = 1000$ ) to the data reported in Figure 4.10b, where the variables  $h_c$  and  $h_j$  are comparable components of the total excavation depth  $h_{ex}$ .

The resulting estimates can be observed in Figure 4.12.

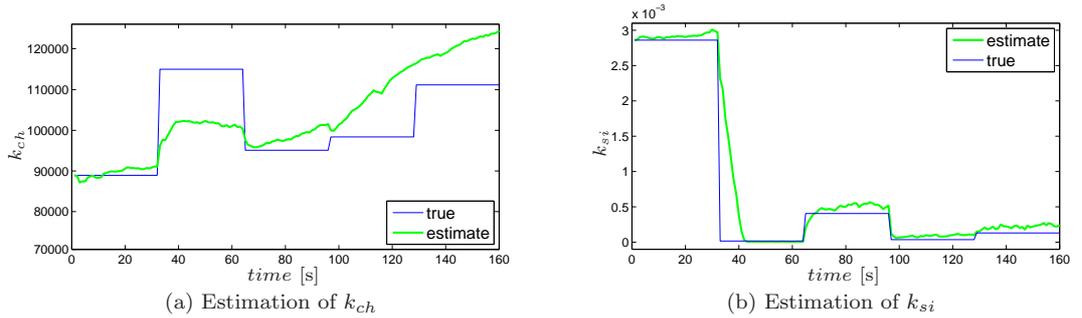


Figure 4.12: Estimation results obtained by PF with use of 1000 particles with resampling (resampling threshold  $N_T = 1000$ ) applied to the data presented in Figure 4.10b. The thick line shows the average of ten independent runs of PF.

We can see that in this case the estimate of  $k_{ch}$  is somewhat better when compared to the estimate given in Figure 4.11 but still unsatisfactory. This slight improvement in the estimation of  $k_{ch}$  comes with the price of less accurate estimate of  $k_{si}$ .

#### 4.4. Solutions to the Cutting and Jetting Estimation Problem

##### Results Obtained by the Particle Filter for Correlated $k_{ch}$ and $k_{si}$

We have applied two PFs, each based on the model (4.36) with different function  $f_{p,a}$  to the simulated data presented in Figure 4.10a. The PFs are using 1000 particles which are resampled at every time step (the resampling threshold  $N_T = 1000$ ).

The estimates obtained by the PF based on the model (4.36) using function  $f_p$  (4.34) are presented in Figure 4.13. The results obtained by the PF based on the model (4.36) using function  $f_a$  (4.35) are presented in Figure 4.14.

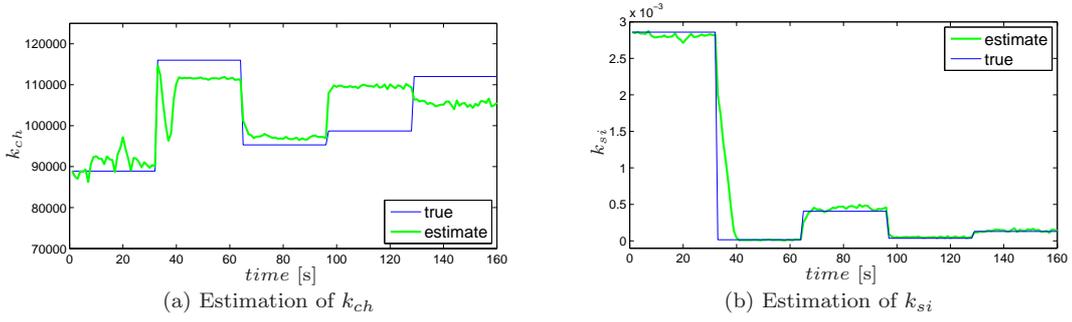


Figure 4.13: Estimation results obtained by PF with use of 1000 particles with resampling (resampling threshold  $N_T = 1000$ ) applied to the data presented in Figure 4.10a. The thick line shows the average of ten independent runs of PF.

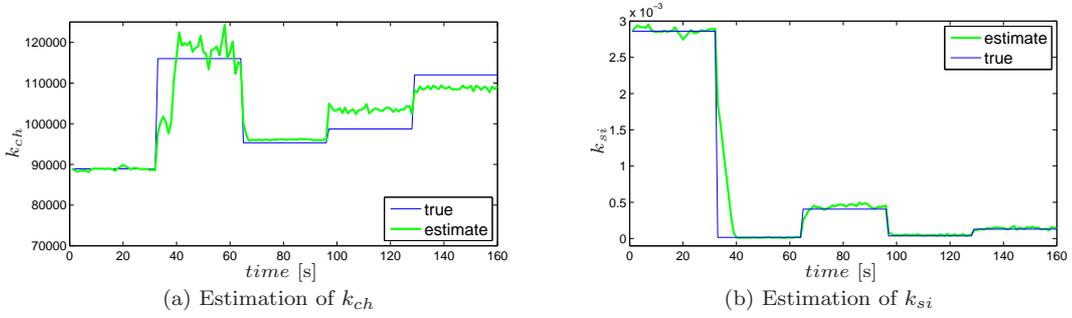


Figure 4.14: Estimation results obtained by PF with use of 1000 particles with resampling (resampling threshold  $N_T = 1000$ ) applied to the data presented in Figure 4.10a. The thick line shows the average of ten independent runs of PF.

We see from Figure 4.13 and Figure 4.14 that the PF that incorporates the knowledge of correlation between the parameters  $k_{si}$  and  $k_{ch}$  through the polynomial function  $f_p$  (4.34) or through the piecewise linear function  $f_a$  (4.35) perform better than the PF that treat these two variables as uncorrelated random walks. Nevertheless, the estimates of the horizontal cutting force coefficient  $k_{ch}$  is still unsatisfactory.

##### Results Obtained by the Cascaded Filter for Correlated $k_{ch}$ and $k_{si}$

We have applied the cascaded filter (Figure 4.9) to the simulated data presented in Figure 4.10a. The PF embedded in the cascaded filter is using 1000 particles, which are resampled at every

## 4. SOLUTIONS TO THE DRAG-HEAD ESTIMATION PROBLEMS

time step (the resampling threshold  $N_T = 1000$ ). The BF embedded in the cascaded filter uses the Gaussian prior  $\pi_0$  with mean given by  $f_a(\hat{k}_{si}^s)$  and standard deviation  $\sigma_0 = 10^4$ . The likelihood  $\mathbb{P}(Y|k_{ch}, U)$  is also Gaussian with mean  $u_c k_{ch}^{-1} + u_j (\hat{k}_{si}^s)^p$  and standard deviation  $\Sigma_{hex}^o$  defined in (4.33).

The estimates obtained by the cascaded filter are presented in Figure 4.15.

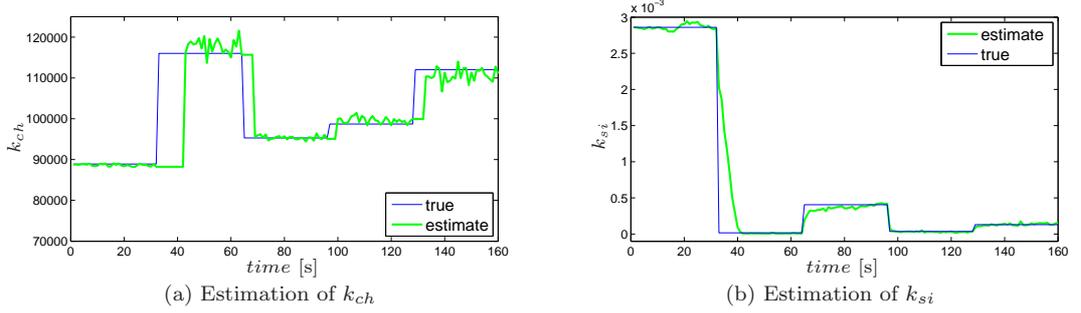


Figure 4.15: Estimation results obtained by the cascaded filter composed of the PF that uses 1000 particles with resampling (resampling threshold  $N_T = 1000$ ), the SSI filter and the BF applied to the data presented in Figure 4.10a. During the transient time we freeze the estimate of  $k_{ch}$  until a new steady state is reached. The thick line shows the average of ten independent filtering runs.

The results presented in Figure 4.15 suggest that the cascaded filter provides estimates that are more accurate than the estimates obtained by the PFs presented in previous sections. On the other hand, due to the SSI filter embedded into the cascaded filter the estimates of  $k_{ch}$  can be produced only after a steady state is reached (by the estimate  $\hat{k}_{si}$ ).

### 4.4.3 Discussion

In this section we have provided a solution to the Cutting and Jetting Estimation Problem formulated in Chapter 2. Due to the highly nonlinear character of the system under consideration, to solve the estimation problem we have employed PF. We have investigated two methods that differ in the way the correlation between the parameters  $k_{ch}$  and  $k_{si}$  is handled.

In the first, the simplest approach, the filter uses model (4.31) where the horizontal cutting force coefficient  $k_{ch}$  and the in situ permeability  $k_{si}$  evolve as independent random walks. The estimates obtained by such a filter are presented in Figure 4.11 and Figure 4.12. The algorithm detects precisely the moments the excavated soil changes and accurately estimates the true state of the parameter  $k_{si}$ . Unfortunately the estimates of the parameter  $k_{ch}$  are very inaccurate, which makes the real life applicability of this filter questionable.

In the second method, the filter uses the more complicated model (4.36). This PF takes advantage of the knowledge of correlation between the parameters  $k_{si}$  and  $k_{ch}$ , which are modeled either by the polynomial function  $f_p$  (4.34) or by the piecewise linear function  $f_a$  (4.35). The results of the simulations suggest that both methods outperforms the PF described by (4.31) that treat these two variables as uncorrelated random walks. The PF using the piecewise linear fit  $f_a$  (4.35) produces more accurate estimates (see Figure 4.14) than the PF using the polynomial fit  $f_p$  (4.34) (see Figure 4.13). Nevertheless, the estimates of the horizontal cutting force coefficient  $k_{ch}$  are still not accurate enough.

The accuracy of the estimates of  $k_{ch}$  can be improved by applying the cascaded filter depicted in Figure 4.9. The results for this case are presented in Figure 4.15 from which we see that the cascaded filter outperforms the remaining methods in terms of accuracy of the estimates. On the downside, due to the SSI filter embedded into the cascaded filter, the estimate of  $k_{ch}$  can be obtained only after the estimate  $\hat{k}_{si}$  reaches steady state. This, however, is not a serious obstacle of the discussed method because the PF embedded into the cascaded filter gives an estimate  $\hat{k}_{si}$  that very quickly settles closely to the true state of  $k_{si}$ , allowing the BF to obtain an accurate estimate of  $k_{ch}$ .

As a final remark note that we assumed that within every time step we are able to collect  $N$  measurements of  $u_c, u_j, h_{ex}$ . The accuracy of the estimation increases with  $N$ . If  $N$  is assumed to be large enough at each window it is possible to obtain a solution to (4.30) by standard least squares or total least squares methods [Huffel and Vandewalle, 1991]. Then, there is no need to employ a dynamic filter of the kind described in this section. However, for  $N = 10$ , as in the case study discussed above, the number of measurements is not sufficient enough to obtain reliable estimates by either of these methods.

## 4.5 Conclusions

In this chapter we have discussed the solutions to the Drag-Head estimation problems that were formulated in Chapter 2. These are the Cutting Estimation Problem and the Cutting and Jetting Estimation Problem. Each of these problems is discussed separately and for each of them we have tested several distinct filtering methods in order to select the most suitable one. Both estimation problems share a common feature of time-varying delay in the measurement of the incoming density  $\rho_i$ . The problem of handling such a delay is independent of the filtering problem and is discussed separately. The block algorithm connecting the delay problem with the filtering problem is presented in Figure 4.3.

The algorithms presented in this chapter aim to estimate soil-dependent parameters such as the ratio  $k_{vh}$  between cutting forces (Cutting Estimation Problem), the horizontal cutting force coefficient  $k_{ch}$  (Cutting Estimation Problem, Cutting and Jetting Estimation Problem) and the in situ permeability  $k_{si}$  (Cutting and Jetting Estimation Problem). The Cutting Estimation Problem is defined for any cutting tool whereas the Cutting and Jetting Estimation Problem is applicable only for tools equipped with cutting and jetting components such as the Drag-Head. In case of the Drag-Head one can switch from one estimation problem to an other simply by turning on/off the jets. Because of that it is possible to commence the excavation process using only cutting. Then, we can quickly obtain an accurate estimate of the  $k_{vh}$  by solving the Cutting Estimation Problem with any of the parametric filters discussed presented in Section 4.3. Then we can turn on the jets to boost the production and solve the Cutting and Jetting Estimation Problem using the previously obtained estimate of  $k_{vh}$  as an input to the system.

To find a solution to the Cutting Estimation Problem we have investigated several nonlinear parametric filters. These are: EKF, UKF, CDF, GSF. The simulation results presented in Figures 4.4, 4.5, 4.6, and 4.7 suggest that each of the discussed nonlinear filters gives an adequate solution to the Cutting Estimation Problem. This is confirmed by Table 4.2 which compares the convergence time, the overshoot, and the MSE of the estimates of  $k_{vh}$  obtained by the aforementioned filters applied to the simulated data. The EKF, the UKF, the CDF with parameter  $h = 1$ , and the two and three-term GSFs produce estimates that are almost indistinguishable among each other.

The estimates with the lowest MSE are produced by the CDF with parameters  $h = 10^{-1}$ , and  $h = 10^4$ . Furthermore, an appropriately tuned CDF is the fastest to reach a steady state when the excavated soil type changes. The four-term GSF has the fastest convergence time

## 4. SOLUTIONS TO THE DRAG-HEAD ESTIMATION PROBLEMS

---

in the first phase of the estimation where the filters have to counteract the uncertainty in the initial state of the system, which results in a large offset. This is possible because the structure of the GS approximation of the state uncertainty allows the GSF to cover the uncertainty region with higher precision than single Gaussian approximations (EKF, UKF, CDF). This, however, comes with the price of higher computational complexity when compared to the very simple and fast CDF. Thus, when the uncertainty in the initial states is small, e.g., we know the exact position of the system, the CDF is recommended over the GSF (and the remaining EKF and UKF). Note that in general such small uncertainty cannot be guaranteed.

We have tested several nonlinear nonparametric filters to determine the best solution to the Cutting and Jetting Estimation Problem. Numerical simulations showed that exploiting the correlation between the soil-dependent parameters: the horizontal cutting force coefficient  $k_{ch}$  and the in situ permeability  $k_{si}$ , is of vital importance in design of accurate estimator. The simulations suggest that the best results are obtained by the cascaded filter which first employs the PF to obtain an estimate of  $k_{si}$  which is further filtered by the SSI filter, and finally by the BF which produces an estimate of  $k_{ch}$ .

### Further Research

This chapter provides a comprehensive analysis of the Drag-Head estimation problems and the corresponding solutions. Nevertheless, there are several points that have not been discussed in details in this chapter. These are listed below.

- I. *Combining the Cutting Estimation Problem with the Cutting and Jetting Estimation Problem.* Separately, under the corresponding assumptions, each problem has been solved. By combining solutions of both problems into a single algorithm we could potentially relax some of the assumptions imposed on the filters discussed in this chapter. This could be achieved e.g., by designing a control strategy switching between the Cutting mode and the Cutting and Jetting mode discussed in the first part of this section.
- II. *Uncertainty in input variables.* In our investigations several times we have assumed that the uncertainty in certain input variables is negligible. This, however does not always need to be an accurate model of reality. Thus, further investigation of this subject can be of great value for practitioners.
- III. *Optimal prior for the BF.* In the implementation of the cascaded filter for the Cutting and Jetting Estimation Problem we have experimentally tuned the prior of the embedded BF to be a Gaussian with mean  $f_a(\hat{k}_{si}^s)$  and standard deviation  $\sigma_0 = 10^4$ . Further research on the optimal choice of the prior distribution might improve the overall performance of the filter.
- IV. *Correlation between the parameters  $k_{ch}$  and  $k_{si}$ .* In Section 4.4 we have discussed three types of functional relations between the horizontal cutting force coefficient  $k_{ch}$  and the in situ permeability  $k_{si}$ . These were derived to match the empirical data reported in Table 4.3. However, as these are only empirical approximations describing a complex relation between soil-dependent parameters, the robustness of the cascaded filter with respect to the functions  $f_{p,a}$  should be further investigated. Complete robustness study would answer the question whether an accurate estimate of  $k_{ch}$  can be obtained without more prior knowledge on the correlations between different soil-dependent parameters.

Deeper investigation of the topics listed above can potentially lead to further improvements of the estimation methods discussed in this chapter. It will also boost the understanding of the Drag-Head excavation process.

## Chapter 5

# Saturated Particle Filter

Parts of this chapter were published in:

- “*Saturated Particle Filter*”, Proceedings of the American Control Conference, San Francisco 2011, pp. 1819-1824, Paweł Stano, Zsófia Lendek and Robert Babuška.
- “*Convex Saturated Particle Filter*”, Paweł Stano, Zsófia Lendek, Robert Babuška and Arnold J. den Dekker, in press.

### Abstract

In many stochastic dynamical systems the state variables are constrained or saturated which often makes them difficult to estimate by standard filtering methods. To estimate the states of such systems, constrained particle filters have been used with some success. In this chapter we show how the performance of the standard particle filters can be improved if the measurement information is used during the importance sampling of the filtering phase. First, in Section 5.1 the general estimation problem for the SSDS is formally introduced. Next, we show how the estimates can be improved by incorporating the measurements into the filtering algorithm through a user-specified detection function, which aims to detect the saturation as it occurs. The algorithm derived from the aforementioned principle is called the *Saturated Particle Filter* (SPF). Section 5.2 derives a complete SPF framework for a class of systems with one-dimensional constraints. Next, in Section 5.3 the *Convex Saturated Particle Filter* (CSPF), which extends the SPF method to multidimensional systems with convex constraints, is derived.

The effectiveness of the proposed method is demonstrated using two illustrative examples. First, in Section 5.2.3 the SPF, is applied to a one-dimensional Lindley-type stochastic process that depends on an exogenous parameter. Next, in Section 5.3.3, the CSPF is tested on a two-dimensional system that models the motion of an object under random, bounded disturbances. In both cases the simulations show that the proposed filters achieve better performance than the standard *Constrained Bootstrap Particle Filter* (CBPF). Furthermore, the SPF and the CSPF achieve high accuracy using relatively few particles, thus preserving low computational complexity.

### 5.1 Introduction

We start this chapter by explaining the importance of constrained and saturated dynamical systems in the context of dynamic nonparametric filtering. Although in recent years these types

## 5. SATURATED PARTICLE FILTER

---

of systems have received growing attention in the scientific community, still, for a researcher, the field seems largely unexplored. Intrigued by this, we set ourselves the objective of investigating the structural properties of such systems and how they can be used. We discovered that the specific form of saturated systems can be exploited in a way that yields an effective filtering method. This new algorithm, which we called the *Saturated Particle Filter* (SPF), is derived in the current chapter.

### 5.1.1 Motivation

Recall from Chapter 3.4 that the *Particle Filter* (PF) is a non-parametric method for solving a general filtering problem for a *Stochastic Dynamical System* (SDS). This is achieved by estimating a *probability density function* (PDF) of the state rather than a point statistic of the state. Namely, the PF approximates a PDF of the state by a set of points which are obtained by utilizing the Importance Sampling method [Arulampalam et al., 2002; Ristic et al., 2004a], and then weighed according to Bayes' rule. The PF is based on Monte Carlo approximation and it has been proven [Cristian and Doucet, 2002] that, under mild technical assumptions, the estimated PDF converges to the true posterior PDF as the number of samples grows. However, for highly nonlinear and non-Gaussian systems, the PF might require a large number of samples to achieve an accurate estimate. This makes the algorithm computationally expensive, especially in high-dimensional systems, and as a consequence, it limits its on-line applicability. It has been noted [Arulampalam et al., 2002; Carlin et al., 1992; Gilks and Berzuini, 2001; Ristic et al., 2004a] that the choice of the importance sampling density is a crucial step towards reducing the computational costs of the PF, and therefore making the method more feasible for on-line applications.

The properties of the PF have been extensively studied in recent years [Arulampalam et al., 2002; Cristian and Doucet, 2002; Ristic et al., 2004a], and many versions of the PF have been developed for specific types of problems [Arulampalam et al., 2002; Lang et al., 2007; Prakash et al., 2010; Shao et al., 2010]. In particular, state estimation of a *Constrained Stochastic Dynamical System* (CSDS) attracted much attention [Kyriakides et al., 2005; Lang et al., 2007; Prakash et al., 2010; Shao et al., 2010; Stano et al., 2011; Straka et al., 2011]. The CSDS is a system for which, at each time step  $k$ , at least one of the state variables is restricted to a compact set. These systems are frequently met both in industrial applications [Stano et al., 2010; Vachhani et al., 2006], and in theoretical research [Shao et al., 2010; Stadjje, 1997; Straka et al., 2011]. To estimate the states of a CSDS one can use the constrained PF [Kyriakides et al., 2005; Lang et al., 2007; Prakash et al., 2010; Shao et al., 2010]. This method produces a state estimate that does not violate the physical constraints of the system. This is achieved either by discarding unsuitable particles [Kyriakides et al., 2005; Lang et al., 2007], or by projecting them on the boundary of the constraint region [Prakash et al., 2010; Shao et al., 2010]. The latter approach is especially suitable for systems characterized by PDFs that are singular (discontinuous) at the boundary of the constraint region. The boundary of such a set is called the saturation region and the particles located at this boundary are called the saturated particles. A system defined by this type of PDF is called a *Saturated Stochastic Dynamical System* (SSDS), which is a special class of the CSDS.

### 5.1.2 Generic Saturated Stochastic Dynamical System

The goal of this section is to present the generic mathematical framework that is used to model a saturated dynamical system. First, let us recall that the systems under consideration are

defined by

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k), \quad (5.1a)$$

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k, \mathbf{v}_k), \quad (5.1b)$$

$$\mathbf{x}_0 \sim p_0(\cdot). \quad (5.1c)$$

where  $\mathbf{w}_k$  and  $\mathbf{v}_k$  are mutually independent random variables that take values in the state space  $\mathcal{X}$  and the observation space  $\mathcal{Y}$  respectively,  $\mathbf{f}_k : \mathcal{X} \times \mathcal{U} \times \mathcal{X} \rightarrow \mathcal{X}$  is a (possibly nonlinear) function that describes the state evolution,  $\mathbf{h}_k : \mathcal{X} \times \mathcal{Y} \rightarrow \mathcal{Y}$  is a (possibly nonlinear) function that establishes the observation model,  $\mathbf{u}_k$  denotes the deterministic input at time step  $k$ , and  $p_0$  is a PDF of the initial state  $\mathbf{x}_0$ . Then, the *Stochastic Dynamical System* (SDS) is defined as the sequence of pairs  $\{(\mathbf{x}_k, \mathbf{y}_k)\}_{k=0}^{+\infty}$  composed of discrete-time stochastic processes  $\{\mathbf{x}_k\}_{k=0}^{+\infty}$  and  $\{\mathbf{y}_k\}_{k=0}^{+\infty}$ .

A saturated system is obtained by imposing extra conditions on the system defined by (5.1). Namely, we consider systems that satisfy the following definition:

**Definition 5.1** (Saturated Stochastic Dynamical System). *Let  $\{(\mathbf{x}_k, \mathbf{y}_k)\}_{k=0}^{+\infty}$  be a SDS defined by (5.1). The sequence of tuples  $\{(\mathbf{x}_k, \mathbf{y}_k)\}_{k=0}^{+\infty}$  is called a Saturated Stochastic Dynamical System (SSDS) if for each  $k \geq 1$  there exists a compact set  $A \in \mathcal{X}$  such that:*

- I. *the support of the conditional distribution of the state  $\mathbf{x}_{k+1}$  given  $\mathbf{x}_k$ ,  $\mathbf{u}_k$ , and  $\mathbf{y}_{k+1}$  is contained within  $A$ , i.e.,*

$$\mathbb{P}(\mathbf{x}_{k+1} \in A | \mathbf{x}_k, \mathbf{u}_k, \mathbf{y}_{k+1}) = 1, \quad (5.2)$$

- II. *the transition probability of the state  $\mathbf{x}_{k+1}$  belonging to  $\partial A$ , given  $\mathbf{x}_k$  and  $\mathbf{u}_k$ , is positive, i.e.,*

$$\mathbb{P}(\mathbf{x}_{k+1} \in \partial A | \mathbf{x}_k, \mathbf{u}_k) > 0, \quad (5.3)$$

where  $\partial A$  denotes the boundary of the set  $A$ .

In what follows, we restrict our considerations to SSDSs with measurements corrupted with additive zero-mean noise. From the engineering point of view this assumption is not very restrictive as such systems are commonly met in practical applications. The observation model of such a SSDS is given by

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{v}_k, \quad (5.4)$$

where  $\mathbb{E}\mathbf{v}_k = 0$ .

## 5.2 One-Dimensional Saturated Particle Filter

This section begins with a formal definition of the class of systems that are analyzed, i.e., the one dimensional SDS. For such one-dimensional SDSs the new PF is derived. The multidimensional extension of the new algorithm is discussed in Section 5.3.

### 5.2.1 One-Dimensional Saturated Stochastic Dynamical System

For this section we assume that  $\{x_k\}_{k=0}^{+\infty}$  and  $\{y_k\}_{k=0}^{+\infty}$  are one-dimensional real-valued processes. Furthermore, we consider only one-dimensional deterministic inputs  $u_k \in \mathbb{R}$ . Finally, it is assumed that the process  $\{x_k\}_{k=0}^{+\infty}$  is non-negative, i.e.,  $x_k \geq 0$  for every  $k \geq 0$ . By considering only systems bounded on the positive real line  $\mathbb{R}_+$  we greatly simplify the calculations that

## 5. SATURATED PARTICLE FILTER

---

follow. However, it has to be noted that such a condition is of purely technical nature. Indeed, it holds true that any variable bounded in  $\mathbb{R}$ , by an affine change of variables, can be transformed into a variable constrained in  $\mathbb{R}_+$ .

In case of one-dimensional systems it is convenient to consider dynamical systems characterized by the minimum function as it is described in Definition 5.2.

**Definition 5.2** (One-Dimensional Saturated Stochastic Dynamical System). *Let us consider a real-valued SDS for which (5.4) is satisfied. Furthermore, let us assume that there exist a function  $C : \mathbb{R}_+ \times \mathbb{R} \rightarrow \mathbb{R}_+$  and a function  $\tilde{f}_k : \mathbb{R}_+ \times \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}_+$  such that for each  $k \geq 1$ , the evolution of the states follows:*

$$x_{k+1} = \min \left( \tilde{f}_k(x_k, u_k, w_k), C(x_k, u_k) \right), \quad (5.5)$$

and

$$\mathbb{P}(x_{k+1} = C(x_k, u_k)) > 0. \quad (5.6)$$

The function  $\tilde{f}_k$  from Definition 5.2 corresponds to the transition function of the “unsaturated” system, i.e., the system with the state that is not bounded by the function  $C$ . Note that by Definition 5.1 the bounds  $\{C(x_k, u_k)\}_{k=0}^{+\infty}$  associated with such a SSDS form a (possibly unbounded) stochastic process. Possible realizations of the stochastic processes  $\{x_k\}_{k=0}^{+\infty}$  and  $\{C(x_k, u_k)\}_{k=0}^{+\infty}$  are illustrated in Figure 5.1.

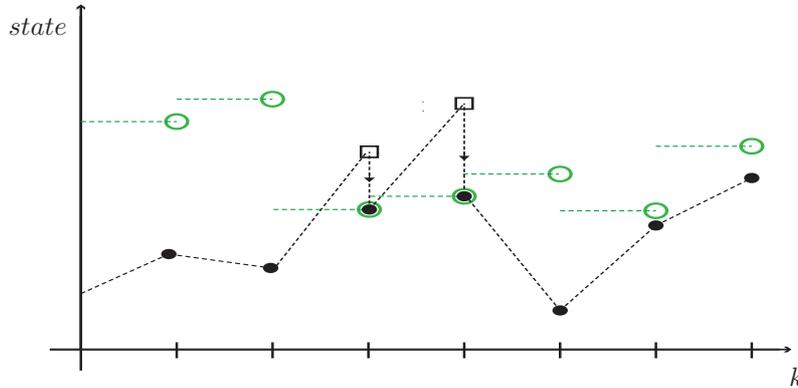


Figure 5.1: Trajectories of the saturated system  $\{x_k\}_{k=0}^{+\infty}$  (small filled circles) and their bounds  $\{C(x_k, u_k)\}_{k=0}^{+\infty}$  (large empty circles). When the unsaturated variable  $\tilde{f}_k(x_k, u_k, w_k)$  (empty squares) exceeds the saturation bound  $C(x_k, u_k)$  (horizontal dotted lines) it is projected onto the appropriate bound (vertical dotted lines). In such cases the realizations of the processes  $\{x_k\}_{k=0}^{+\infty}$  and  $\{C(x_k, u_k)\}_{k=0}^{+\infty}$  are overlapping (small circles within large circles).

For systems with a continuous state space it is reasonable to assume that for every time step  $k$  the random variable  $\tilde{f}_k(x_k, u_k, w_k)$  has a continuous PDF. This, however, does not hold for  $x_{k+1}$ . Indeed, from (5.6) it follows that each variable  $x_{k+1}$  has a singularity at the point  $C(x_k, u_k)$ . This means that the PDF of  $x_{k+1}$  is continuous up to the point  $C(x_k, u_k)$  in which a positive probability mass is focused. Therefore, the conditional density of the

## 5.2. One-Dimensional Saturated Particle Filter

---

variable  $x_{k+1}$ , given the previous state  $x_k$  and the deterministic input  $u_k$ , is given by:

$$\mathbb{P}(x_{k+1} = x | x_k, u_k) = \mathbb{P}\left(\tilde{f}_k(x_k, u_k, w_k) = x | x_k, u_k\right) \mathbf{1}_{[0, C(x_k, u_k)]}(x) \quad (5.7a)$$

$$+ \int_{C(x_k, u_k)}^{+\infty} \mathbb{P}\left(\tilde{f}_k(x_k, u_k, w_k) = z | x_k, u_k\right) dz \delta_{C(x_k, u_k)}(x), \quad (5.7b)$$

where  $\mathbf{1}_{[0, C(x_k, u_k)]}$  is an indicator function on the interval  $[0, C(x_k, u_k)]$ , and  $\delta_{C(x_k, u_k)}$  is the Dirac delta function centered at the point  $C(x_k, u_k)$ . The PDF of such a variable is illustrated in Figure 5.2.

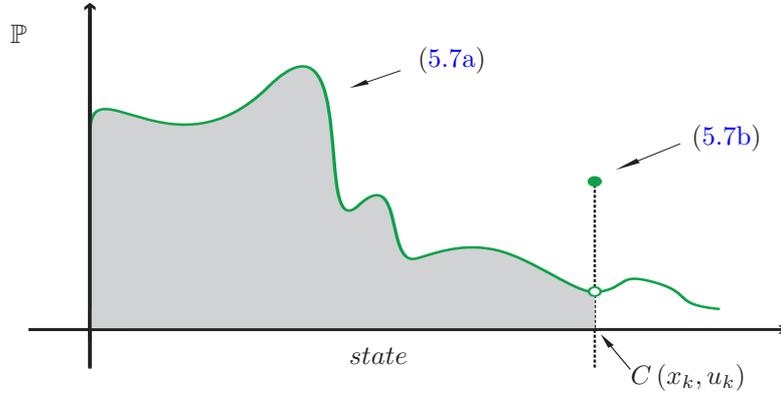


Figure 5.2: The PDF of the saturated variable  $x_{k+1}$  given the past state  $x_k$ . The PDF is composed of a continuous part (5.7a) and a singular mass (5.7b) concentrated at the saturation point  $C(x_k, u_k)$ .

An illustrative example of SSDS will also be discussed in detail in Section 5.2.3. To keep the calculation that follow simple, in Definition 5.1 we chose the state variables  $x_{k+1}$  to be bounded by the interval  $[0, C(x_k, u_k)]$  that lays on the positive real line  $\mathbb{R}_+$ .

Having the SSDS defined in such a way, we are interested in estimating the true state  $x_k$  of the system from the available measurements  $y_k$ .

### 5.2.2 One-Dimensional Saturated Particle Filter

In this section we introduce the *Saturated Particle Filter* (SPF), the estimation algorithm that is designed specifically for the class of one-dimensional SSDS discussed in Section 5.2.1. The general idea behind the SPF presented in current section can be further extrapolated to higher dimensional systems, provided certain technical conditions hold. Such a multidimensional extension of the SPF algorithm is discussed in detail in Section 5.3.

We begin with describing the general idea of the proposed method. The SPF is a *Sequential Importance Sampling* (SIS) type algorithm that samples particles  $\{x_{k+1}^i\}_{i=1}^N$  from a special importance kernel  $\tilde{K}(\cdot | x_k^i, u_k, y_{k+1})$  that is chosen in such a way that the distribution of these particles is “closer” to the true posterior PDF  $\pi_{k+1|k+1}$  than the distribution of the particles obtained by the *Bootstrap Particle Filter* (BPF), i.e., by sampling from the transition probability kernel  $K(\cdot | x_k^i, u_k)$ . The SPF’s improved estimation performance comes from the use of the *detection function* whose purpose is to quickly detect whether saturation occurred by comparing the measurements with the state constraints. Namely, the detection function is intended as

## 5. SATURATED PARTICLE FILTER

---

a “pseudometric” between  $y_{k+1}$  and  $h_{k+1}(C(x_k^i, u_k))$ . This extra information is used to force the particles to move to the “appropriate” region already in the sampling step of the algorithm.

For the class of systems presented in Definition 5.2 let us introduce:

**Definition 5.3** (One-dimensional detection function). *Let  $\alpha : \mathbb{R} \rightarrow \mathbb{R}$  be a function for which the following conditions are satisfied:*

- I.  $\alpha$  is non-decreasing,
- II. there exists  $y_0 \in \mathbb{R}$  such that  $\alpha(y_0) = 0$ .

Then, the mapping

$$(y_{k+1}, x_k^i) \mapsto \alpha(y_{k+1} - h_{k+1}(C(x_k^i, u_k))) \quad (5.8)$$

is called a detection function.

In general, the choice of detection function depends on the dynamics of the system under consideration. For the clarity of the argument, throughout this section we assume that the detection function  $\alpha$  is specified by the user.

The detection function, as it is shown in what follows, by comparing the measurements with the state constraints updates the probability of saturation. This information is used to force the particles to move to the appropriate region. Definition 5.3 is not the only possible way of defining detection functions. Nevertheless, for the moment we use this definition because for the one-dimensional SSDS it illustrates well the idea behind the SPF. Since there is a one to one relation between the detection function defined by (5.8) and  $\alpha$ , in what follows, we do not make a distinction between these two objects.

Let us consider the SSDS defined by (5.5)–(5.6). Furthermore, let  $\{(x_k^i, \omega_k^i)\}_{i=1}^N$  be the approximation of the updated density of the process at time step  $k$ . We say that the particle  $x_{k+1}^i$  is saturated if  $x_{k+1}^i$  is projected onto  $C(x_k^i)$ , which is equivalent to the projection method described in [Shao et al., 2010]. Indeed, it makes no difference whether the ‘bad’ particles drawn from an unconstrained continuous distribution are projected on the saturation point, or each particle is set to the saturation point with the probability of saturation. The resulting sets of particles are indistinguishable in a statistical sense. Then, for each  $i \in \{1, \dots, N\}$ , given the previous particle  $x_k^i$ , the probability that the particle  $x_{k+1}^i$  will saturate follows by (5.7b):

$$\mathbb{P}(x_{k+1}^i = C(x_k^i, u_k)) = \int_{C(x_k^i, u_k)}^{+\infty} \mathbb{P}(\tilde{f}_k(x_k, u_k, w_k) = z | x_k^i, u_k) dz. \quad (5.9)$$

For the convenience of the derivations that follow we define the probabilities of saturation:

**Definition 5.4** (Probabilities of saturation). *For every  $i = 1, \dots, N$  the predicted probability of saturation  $q_i$  is given by*

$$q_i = \int_{C(x_k^i, u_k)}^{+\infty} \mathbb{P}(\tilde{f}_k(x_k, u_k, w_k) = z | x_k^i, u_k) dz, \quad (5.10)$$

and the updated probability of saturation  $q_i^\alpha$  is given by

$$q_i^\alpha := \begin{cases} 1 & \text{if } q_i + \alpha(y_{k+1} - h_{k+1}(C(x_k^i))) > 1, \\ 0 & \text{if } q_i + \alpha(y_{k+1} - h_{k+1}(C(x_k^i))) < 0, \\ q_i + \alpha(y_{k+1} - h_{k+1}(C(x_k^i))) & \text{otherwise.} \end{cases} \quad (5.11)$$

---

## 5.2. One-Dimensional Saturated Particle Filter

Note that the probability  $q_i^\alpha$  can be defined in an alternative, more concise fashion:

$$q_i^\alpha := \max(\min(q_i + \alpha(y_{k+1} - h_{k+1}(C(x_k^i, u_k))), 1), 0) \quad (5.12)$$

With the updated probability of saturation  $q_i^\alpha$  defined by (5.11), and the detection function  $\alpha$ , we are ready to define the importance density  $\tilde{K}_k$  of the SPF by:

$$\tilde{K}_k(x|x_k^i, y_{k+1}, u_k) := q_i^\alpha \delta(C(x_k^i, u_k) - x) \quad (5.13a)$$

$$+ \frac{1 - q_i^\alpha}{1 - q_i} \mathbb{P}(\tilde{f}_k(x_k, u_k, w_k) = x|x_k^i, u_k) \mathbf{1}_{[0, C(x_k^i, u_k)]}(x), \quad (5.13b)$$

where  $\delta$  denotes the Dirac delta function, and  $\mathbf{1}_{[0, C(x_k^i, u_k)]}$  is an indicator function on the interval  $[0, C(x_k^i, u_k)]$ . It can be easily seen that  $\tilde{K}_k$  defines a probability kernel. Indeed, the operator  $\tilde{K}_k$  is nonnegative and it integrates to one. Note that  $\alpha \equiv 0$  yields  $q_i^\alpha = q_i$  hence the importance density of the BPF is a special case of  $\tilde{K}_k$  with  $\alpha \equiv 0$ .

Given the particle  $x_k^i$ , a new particle  $x_{k+1}^i$  is drawn from the importance density  $\tilde{K}_k(\cdot|x_k^i, u_k, y_{k+1})$ . By (5.13b) the particle  $x_{k+1}^i$  saturates, i.e.,  $x_{k+1}^i = C(x_k^i, u_k)$ , with the probability  $q_i^\alpha$ , and with probability  $1 - q_i^\alpha$  it is drawn from

$$\frac{1}{1 - q_i} \mathbb{P}(\tilde{f}_k(x_k, u_k, w_k) = x|x_k^i, u_k) \mathbf{1}_{[0, C(x_k^i, u_k)]}(x) \quad (5.14)$$

The associated weights  $\omega_{k+1}^i$  are computed using (3.84). Namely, if  $x_{k+1}^i$  saturates, i.e.,  $x_{k+1}^i = C(x_k^i, u_k)$ , then, by the definitions of  $q_i$  and  $q_i^\alpha$ , the weight  $\omega_{k+1}^i$  follows the formula:

$$\omega_{k+1}^i \propto \omega_k^i \frac{q_i}{q_i^\alpha} g_{k+1}(y_{k+1}|x_{k+1}^i, u_k). \quad (5.15)$$

If  $x_{k+1}^i$  does not saturate, the weight  $\omega_{k+1}^i$  is updated by:

$$\omega_{k+1}^i \propto \omega_k^i \frac{1 - q_i}{1 - q_i^\alpha} g_{k+1}(y_{k+1}|x_{k+1}^i, u_k). \quad (5.16)$$

Note that by (5.4) the likelihood function  $g_{k+1}$  defined by (3.79) takes the form:

$$g_{k+1}(y_{k+1}|x_{k+1}^i, u_k) = \mathbb{P}(v_{k+1} = y_{k+1} - h_{k+1}(x_{k+1}^i, u_k)). \quad (5.17)$$

The SPF is summarized in Algorithm 5.1. Note that the updated probability of saturation depends on the choice of the detection function  $\alpha$ . Therefore, through  $q_i^\alpha$ , the SPF also depends on  $\alpha$ .

The proposed SPF combines the prior state particle  $x_k^i$  with the most recent measurement  $y_{k+1}$  to compute the updated probability of saturation  $q_i^\alpha$ . For large values of  $q_i^\alpha$  the algorithm forces the particles to be close to the saturation region, by which we understand the set  $\{C(x^i, u_k)\}$ , whereas for small values of  $q_i^\alpha$  the particles are set further from the saturation region. Figure 5.3 schematically presents the difference between the SPF and the benchmark methods: the *Unconstrained Bootstrap Particle Filter* (UBPF), the *Constrained Bootstrap Particle Filter* (CBPF), i.e., the BPF [Arulampalam et al., 2002] modified by the projection approach of [Shao et al., 2010].

The accuracy of the estimation depends on the detection function, which must be chosen according to the SSDS under consideration.

## 5. SATURATED PARTICLE FILTER

---

**Algorithm 5.1** Saturated Particle Filter for a given  $\alpha$

---

**Input:**  $\{(x_k^i, \omega_k^i)\}_{i=1}^N, y_{k+1}, u_k$

**Output:**  $\{(x_{k+1}^i, \omega_{k+1}^i)\}_{i=1}^N$

**for**  $i = 1, 2, \dots, N$  **do**

    Compute the probability  $q_i$  according to (5.10)

    Compute the probability  $q_i^\alpha$  according to (5.11)

    Draw from the standard uniform distribution  $u \sim \mathcal{U}(0, 1)$

**if**  $u \leq q_i^\alpha$  **then**

        Particle  $x_{k+1}^i$  saturates:

$$\begin{aligned} x_{k+1}^i &:= C(x_k^i, u_k) \\ \omega_{k+1}^i &\propto \omega_k^i \frac{q_i}{q_i^\alpha} g_{k+1}(y_{k+1} | x_{k+1}^i, u_k) \end{aligned}$$

**else**

        Particle  $x_{k+1}^i$  does not saturate:

$$\begin{aligned} x_{k+1}^i &\sim \frac{1}{1 - q_i} \mathbb{P}\left(\tilde{f}_k(x_k, u_k, w_k) = \bullet | x_k^i, u_k\right) \mathbf{1}_{[0, C(x_k^i, u_k))}(\bullet) \\ \omega_{k+1}^i &\propto \omega_k^i \frac{1 - q_i}{1 - q_i^\alpha} g_{k+1}(y_{k+1} | x_{k+1}^i, u_k) \end{aligned}$$

**end if**

**end for**

---

### 5.2.3 Numerical Simulations

In this section we apply the SPF to a system that depends on an external parameter  $\theta$  and allows relatively large measurement noises. We show that with the proper choice of the detection function  $\alpha$ , the SPF outperforms the CBPF filter in tracking rapid changes in the dynamics of the system.

The system used to compare the SPF and the CBPF to the SSDS is given by:

$$x_{k+1} = \min(x_k + w_k, C(x_k)), \quad (5.18a)$$

$$y_k = x_k + v_k, \quad (5.18b)$$

where  $w_k$  is an exponential random variable with parameter  $\theta \cdot C(x_k)$ , i.e., with the expected value  $\mathbb{E}w_k = (\theta \cdot C(x_k))^{-1}$ . The variable  $v_k$  is a zero-mean Gaussian variable with standard deviation  $\sigma_v$ . The system under consideration has no deterministic inputs, hence, throughout this subsection we skip the symbol  $u_k$  from the notation.

The state model (5.18a) is nonlinear and non-Gaussian, whereas the observation model (5.18b) is both linear, and conditionally Gaussian. The stochastic process (5.18a) is a Lindley-type process, i.e., it is a modification of the celebrated *Lindley's recursion* [Lindley, 1952; Stadjé, 1997; Vlasiou et al., 2004]. These types of processes are extensively used in queueing theory [Simonot, 1997; Vlasiou et al., 2004]. The possible interpretation of (5.18a) is that  $x_k$  is the total operation time of a server after the  $k$ -th customer has been served, the random variable  $w_k$  models the service waiting time (hence, it is exponentially distributed), and the bound  $C(x_k)$  gives the maximal time of service after which the server proceeds to the next customer. By choosing this type of systems for our experiments we are able to illustrate the main properties of our

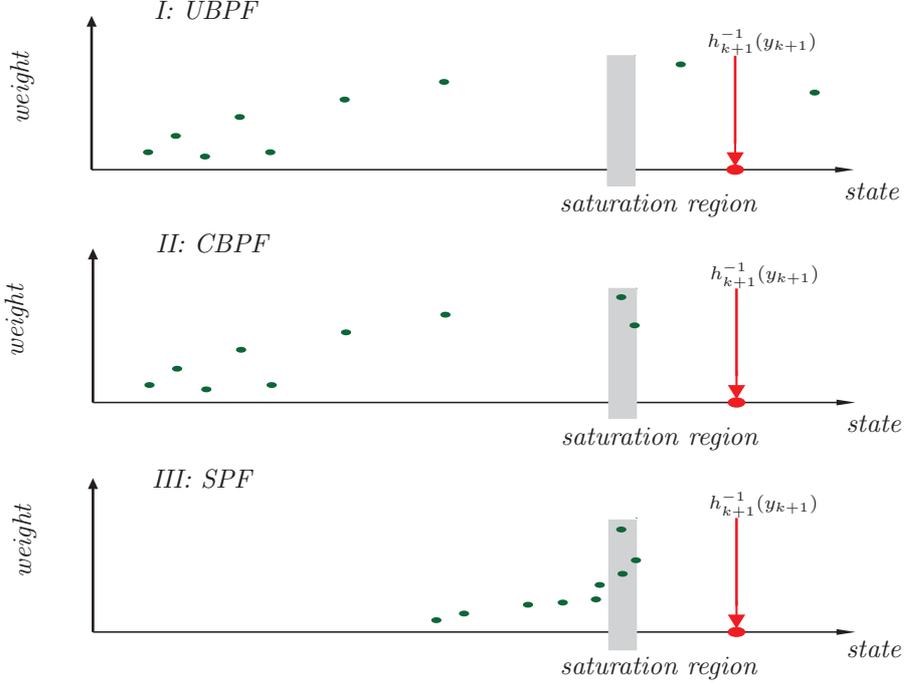


Figure 5.3: Visualization of the distribution of the particles with corresponding weights  $\omega$  obtained by the UBPF (top), the CBPF (middle) and the SPF (bottom). Some of the particles obtained by the UBPF violate the physical constraints on the system (saturation region  $\bigcup_i C(x_k^i, u_k)$ ), others are located far from the actual observation  $h_{k+1}^{-1}(y_{k+1})$ . The CBPF projects the “unphysical” particles onto the saturation region  $\bigcup_i C(x_k^i, u_k)$ , but does not move the remaining particles. The SPF projects the bad particles onto the saturation region  $\bigcup_i C(x_k^i, u_k)$  and forces the remaining particles to concentrate close to the saturation region  $\bigcup_i C(x_k^i, u_k)$ .

algorithm at the same time avoiding expensive numerical computations of the integral (5.29). This is possible because the variable  $w_k$  is exponentially distributed, hence the integral (5.29) can be computed explicitly as:

$$q_k^i = \exp(-\theta C(x_k^i) (C(x_k^i) - x_k^i)). \quad (5.19)$$

The boundary function  $C(\cdot)$  is defined by:

$$C(x) := \begin{cases} x + 4 & \text{if } x < 15, \\ 0.7x + 8.5 & \text{otherwise.} \end{cases} \quad (5.20)$$

To illustrate the capabilities of the proposed filter, starting from the initial state of the system  $x_0 = 7$ , we simulated the evolution of the system (5.18)–(5.20) for 100 time steps. During the first 50 steps, the parameter  $\theta$  is set to 1, during the second 50 steps it is set to  $\frac{1}{30}$ . This models a rapid change in conditions external to the system. In our single-server queue analogy this corresponds to the situation when from a certain time onwards due to e.g., malfunction, the processing speed of a server is significantly (30-fold) reduced.

To simulate a noisy-measurement environment (5.18b), the standard deviation  $\sigma_v$  of the variable  $v_k$  is set to  $\sigma_v = 3$ .

## 5. SATURATED PARTICLE FILTER

---

Figure 5.5 presents two independent simulation runs of the system (5.18)–(5.20) and two filtered signals obtained by the average of ten independent runs of both, the CBPF and the SPF. Both the CBPF and the SPF use the state model (5.18a) with the parameter  $\theta = 1$  for the whole time of the simulation. The initial state  $p_0$  for both filters is equal to  $p_0(\cdot) = \mathcal{N}(\cdot; 7, 1)$  (the PDF of the Gaussian variable with the mean and the standard deviation equal to 7 and 1, respectively). The number of the particles is set to  $N = 100$ , and the resampling threshold is set to  $N_T = 50$ .

In the simulations we tested the SPF with two different detection functions, depicted in Figure 5.4, that are defined by:

- I. For the first SPF the detection function  $\alpha_1$  is defined by:

$$\alpha_1(x) := \begin{cases} \log(x+1) & \text{if } x > 0, \\ -\log(-x+1) & \text{otherwise.} \end{cases} \quad (5.21)$$

The function  $\alpha_1$  is antisymmetric in zero, which means that the probability of saturation  $q_i^\alpha$  is increased or decreased proportionally to the distance between the measurement  $y_{k+1}$  and the saturation bound  $C(x_k^i)$ . If the distance  $|y_{k+1} - C(x_k^i)|$  is greater than ( $\approx$ ) 1.7 then, depending on the sign of the difference, the probability of saturation  $q_i^\alpha$  is equal to zero or to one.

- II. For the second SPF the detection function  $\alpha_2$  is defined as:

$$\alpha_2(x) := \begin{cases} \log(x+1) & \text{if } x > 0, \\ -\log(-x+1) & \text{if } x > -\frac{1}{2}, \\ -3\log(-x+1) + 2\log\left(\frac{3}{2}\right) & \text{otherwise.} \end{cases} \quad (5.22)$$

The function  $\alpha_2$  is not antisymmetric as was  $\alpha_1$ . In this case, when the measurement  $y_{k+1}$  is smaller than  $C(x_k^i) - \frac{1}{2}$ , the probability of saturation  $q_i^\alpha$  decreases much faster with the distance  $|y_{k+1} - C(x_k^i)|$  and reaches zero when  $y_{k+1} < C(x_k^i) - 0.83$ . When the measurement  $y_{k+1}$  is greater than  $C(x_k^i) - \frac{1}{2}$  the probability of saturation  $q_i^\alpha$  is adjusted identically as it was for the function  $\alpha_1$ .

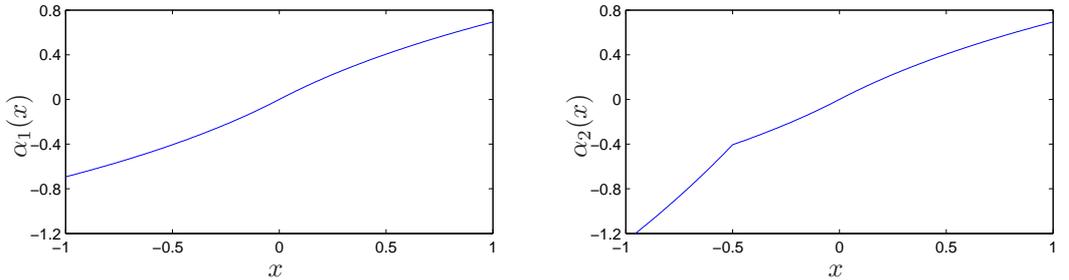


Figure 5.4: The antisymmetric detection function  $\alpha_1$  (left) and the asymmetric detection function  $\alpha_2$  (right).

Figure 5.5 presents the state estimated by the SPF with the antisymmetric detection function  $\alpha_1$  (left) and by the SPF that uses the asymmetric detection function  $\alpha_2$  (right). The estimated signals in Figure 5.5 are computed as the average of ten independent filter runs each applied to the same measurement signal. In each of the parallel runs, for both the CBPF and

the SPF, the estimated value of the state is computed by taking the weighted mean of the particles, i.e.,  $\hat{x}_k = \sum_{i=1}^N \omega_k^i x_k^i$ . This corresponds to the Minimum Mean Square Error (MMSE) estimator [Ristic et al., 2004a].

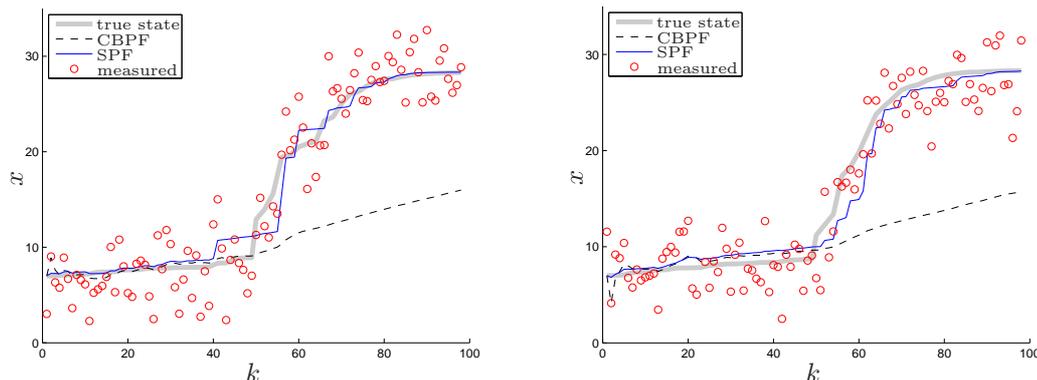


Figure 5.5: The CBPF and the SPF applied to the system (5.18)–(5.20). The thick solid line is the true value of the state, the circles denote the measurements of the system, the thin dashed line denotes the MMSE estimate obtained by the CBPF filter. The thin solid line represents the MMSE estimate of the state obtained by the SPF with the detection function  $\alpha_1$  (left) and  $\alpha_2$  (right) respectively.

The results presented in Figure 5.5 show that both the SPF and the CBPF filter perform similarly during the phase when their state model correspond to the true state process ( $\theta = 1$ ). When the external parameter changes ( $\theta = \frac{1}{30}$ ) the SPF is able to track the true state, whereas the CBPF filter fails to do so. This happens because, as the measurements get further away from what the SPF consider to be the saturation bound for the state, the algorithm puts more “trust” into the observations rather than into the embedded model, which indeed does not match the true model any longer.

The difference in the detection functions  $\alpha_1$  and  $\alpha_2$  does not result in a qualitative change of filtered signal. Thus, the results suggest that the SPF applied to the system (5.18)–(5.20) is robust with respect to the choice of the detection function  $\alpha$ .

### 5.3 Convex Saturated Particle Filter

The SPF proposed in Section 5.2.2 has been derived for a special class of SSDSs, namely for systems that allow only one-dimensional saturation, i.e., if  $\mathbf{x}_k = [x_k(1), \dots, x_k(n)]^T$  is an  $n$ -dimensional state variable then only one of the variables  $x_k(1), \dots, x_k(n)$  can be saturated. Such algorithm can be easily extended to systems with multidimensional saturations provided that the saturated variables are independent. However, the extension for general multidimensional SSDSs is not that straightforward. In this chapter we aim to fill this gap in the SPF framework by deriving the *Convex Saturated Particle Filter* (CSPF), which is applicable to multidimensional systems with convex constraints imposed on the states. The assumption of the convexity of the constraints, from the practical perspective, is not very restrictive. In fact a stronger condition of linear constraints is commonly met in the literature [Dantzig, 1998; Prakash et al., 2010; Straka et al., 2011; Vachhani et al., 2006].

## 5. SATURATED PARTICLE FILTER

---

In Section 5.3.1 the mathematical framework of the Convex Saturated Stochastic Dynamical System is defined and the estimation problem is stated. The novel CSPF is derived in Section 5.3.2. In Section 5.3.3 the new filter is compared with the benchmark CBPF.

### 5.3.1 Convex Saturated Stochastic Dynamical System

For the  $n$ -dimensional state space  $\mathcal{X}$ , the  $m$ -dimensional input space  $\mathcal{U}$ , and the  $p$ -dimensional observation space  $\mathcal{Y}$ , we consider the stochastic dynamical systems characterized by:

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k), \quad (5.23a)$$

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{v}_k, \quad (5.23b)$$

where, for every  $k \geq 1$ ,  $\mathbf{w}_k \in \mathcal{X}$  and  $\mathbf{v}_k \in \mathcal{Y}$  are mutually independent random vectors (possibly non-Gaussian),  $\mathbf{f}_k : \mathcal{X} \times \mathcal{U} \times \mathcal{X} \rightarrow \mathcal{X}$  is a (possibly nonlinear) function that describes the state evolution,  $\mathbf{h}_k : \mathcal{X} \rightarrow \mathcal{Y}$  is a (possibly nonlinear) function that establishes the observation model, and  $\mathbf{u}_k$  denotes the deterministic input at time step  $k$ .

The stochastic process defined by (5.23) is Markovian, which allows for the recursive estimation of the state of the system. In what follows we consider SSDSs such that for each  $k \geq 1$  the constraint region of the variable  $\mathbf{x}_{k+1}$ , dependent on the previous state  $\mathbf{x}_k$  and the previous input  $\mathbf{u}_k$ , is a convex set. The precise conditions that such systems need to satisfy are listed in Definition 5.5.

**Definition 5.5** (Convex Saturated Stochastic Dynamical System). *Let  $\Sigma_{\mathcal{X}}$  be the collection of measurable subsets of  $\mathcal{X}$ . The SSDS  $\{(\mathbf{x}_k, \mathbf{y}_k)\}_{k=0}^{+\infty}$  is called a Convex Saturated Stochastic Dynamical System (CSSDS) if there exist a function  $C : \mathcal{X} \times \mathcal{U} \rightarrow \Sigma_{\mathcal{X}}$  and measurable functions  $\tilde{\mathbf{f}}_k : \mathcal{X} \times \mathcal{U} \times \mathcal{X} \rightarrow \mathcal{X}$  such that for each  $k \geq 1$  the following hold:*

- I. for every convex  $A \in \Sigma_{\mathcal{X}}$  the set  $\mathbf{h}_k(A)$  is convex,
- II. for every  $\mathbf{x} \in \mathcal{X}$  and every  $\mathbf{u} \in \mathcal{U}$  the set  $C(\mathbf{x}, \mathbf{u})$  is precompact [Lee, 2010], open and convex such that  $\mathbf{x} \in C(\mathbf{x}, \mathbf{u})$
- III. for every  $\mathbf{x} \in \mathcal{X}$  and every  $\mathbf{u} \in \mathcal{U}$  the set  $\tilde{\mathbf{f}}(\mathbf{x}, \mathbf{u}, \mathcal{X})$  is convex,
- IV. the state evolution (5.23a) is described by:

$$\begin{aligned} & \text{if } \tilde{\mathbf{f}}_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k) \in C(\mathbf{x}_k, \mathbf{u}_k) \\ & \mathbf{x}_{k+1} = \tilde{\mathbf{f}}_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k) \end{aligned} \quad (5.24a)$$

otherwise

$$\mathbf{x}_{k+1} = \partial C(\mathbf{x}_k, \mathbf{u}_k) \cap \mathcal{R}^0\left(\mathbf{x}_k, \tilde{\mathbf{f}}_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k)\right), \quad (5.24b)$$

where  $\partial C(\mathbf{x}, \mathbf{u})$  denotes the boundary of the set  $C(\mathbf{x}, \mathbf{u})$  and  $\mathcal{R}^0$  is defined by

$$\mathcal{R}^0(\mathbf{x}_k, \mathbf{x}) := \{\mathbf{x}_k + t(\mathbf{x} - \mathbf{x}_k) : t \geq 0\}. \quad (5.25)$$

Note that from the properties of convex sets, the intersection in (5.24b) contains exactly one element. Thus,  $\mathbf{x}_{k+1}$  is uniquely defined. The bounds  $\{C(\mathbf{x}_k, \mathbf{u}_k)\}_{k=0}^{+\infty}$  of a CSSDS form a (possibly unbounded) stochastic process taking values in  $\Sigma_{\mathcal{X}}$ . To help understand the meaning

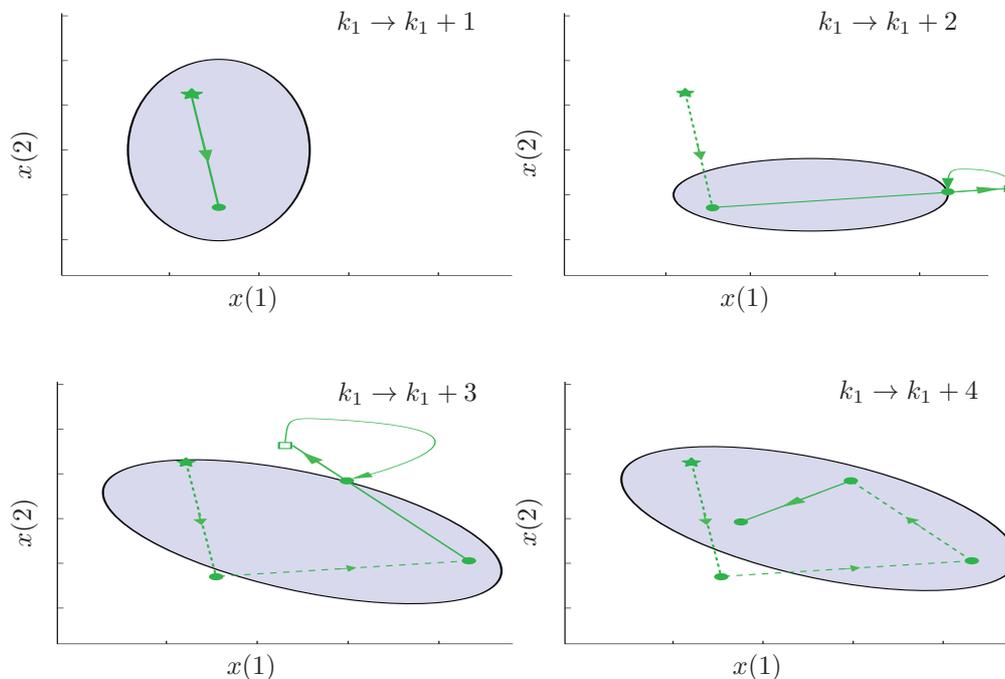


Figure 5.6: Evolution of a two dimensional CSSDS  $\{\mathbf{x}_k\}_{k=0}^{+\infty}$ , from  $\mathbf{x}_{k_1}$  (star) to  $\mathbf{x}_{k_1+4}$  (dots) and its constraint regions  $\{C(\mathbf{x}_k, \mathbf{u}_k)\}_{k=0}^{+\infty}$  (shaded ellipses). When the unsaturated variable  $\tilde{\mathbf{f}}_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k)$  (empty squares) exceeds the saturation boundary  $\partial C(\mathbf{x}_k, \mathbf{u}_k)$  it is projected on the appropriate point (dots) at the saturation boundary.

of (5.24)–(5.25) we have illustrated a possible trajectory of the stochastic processes  $\{\mathbf{x}_k\}_{k=0}^{+\infty}$  and  $\{C(\mathbf{x}_k, \mathbf{u}_k)\}_{k=0}^{+\infty}$  in Figure 5.6.

We are interested in continuous state space, therefore it is reasonable to assume that for every time step  $k$  the random variable  $\tilde{\mathbf{f}}_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k)$  has a continuous PDF. This, however, does not hold for the  $\mathbf{x}_k$ . Indeed, from (5.24) it follows that each variable  $\mathbf{x}_{k+1}$  has singularities at the boundary of  $C(\mathbf{x}_k, \mathbf{u}_k)$ . This means that the PDF of  $\mathbf{x}_{k+1}$  is continuous in the interior of the set  $C(\mathbf{x}_k, \mathbf{u}_k)$  and has discontinuities at the boundary  $\partial C(\mathbf{x}_k, \mathbf{u}_k)$ . This is schematically depicted in Figure 5.8. Nevertheless, the random variable  $\mathbf{x}_{k+1}$  restricted to  $\partial C(\mathbf{x}_k, \mathbf{u}_k)$  has an absolutely continuous distribution with respect to the appropriate Hausdorff measure [Folland, 1984]. To see this, let us first notice that by (5.24)–(5.25) the conditional density of the variable  $\mathbf{x}_{k+1}$  given the previous state  $\mathbf{x}_k$  and input  $\mathbf{u}_k$  is given by:

$$\mathbb{P}(\mathbf{x}_{k+1} = \mathbf{x} | \mathbf{x}_k, \mathbf{u}_k) = \mathbb{P}(\tilde{\mathbf{f}}_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k) = \mathbf{x} | \mathbf{x}_k, \mathbf{u}_k) \mathbf{1}_{C(\mathbf{x}_k, \mathbf{u}_k)}(\mathbf{x}) \quad (5.26a)$$

$$+ \left[ \int_{\mathcal{R}^1(\mathbf{x}_k, \mathbf{x})} \mathbb{P}(\tilde{\mathbf{f}}_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k) = \mathbf{z} | \mathbf{x}_k, \mathbf{u}_k) dH_1(\mathbf{z}) \right] \mathbf{1}_{\partial C(\mathbf{x}_k, \mathbf{u}_k)}(\mathbf{x}), \quad (5.26b)$$

where  $\mathbf{1}_A(\cdot)$  denotes the indicator function of the set  $A$ ,  $H_n$  denotes the  $n$ -dimensional Hausdorff

## 5. SATURATED PARTICLE FILTER

measure, and  $\mathcal{R}^1$  is a set defined by

$$\mathcal{R}^1(\mathbf{x}_k, \mathbf{x}) := \{\mathbf{x}_k + t(\mathbf{x} - \mathbf{x}_k) : t \geq 1\}. \quad (5.27)$$

The PDF of  $\mathbf{x}_{k+1}$  restricted to  $\partial C(\mathbf{x}_k, \mathbf{u}_k)$  is given by (5.26b) which is a continuous function for  $\mathbf{x} \in \partial C(\mathbf{x}_k, \mathbf{u}_k)$ .

To illustrate the relations between the sets  $\mathcal{R}^0$  and  $\mathcal{R}^1$ , Figure 5.7 depicts  $\mathcal{R}^0(\mathbf{x}_k, \tilde{\mathbf{f}}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k))$  and  $\mathcal{R}^1(\mathbf{x}_k, \mathbf{x})$  for a given convex set  $C(\mathbf{x}_k, \mathbf{u}_k)$ .

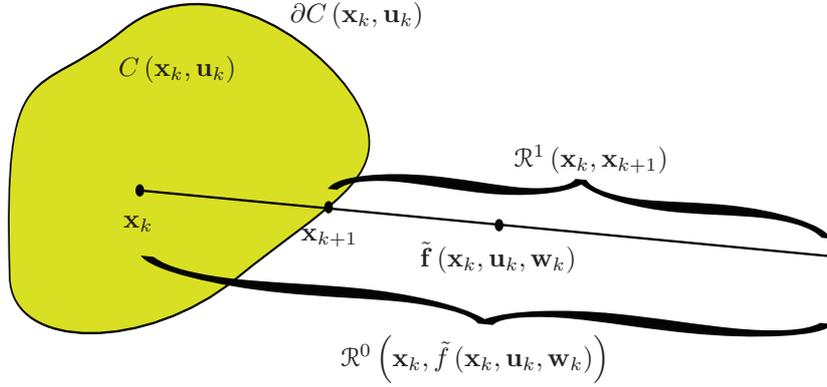


Figure 5.7: The relation between  $\mathcal{R}^1(\mathbf{x}_k, \mathbf{x}_{k+1})$  and  $\mathcal{R}^0(\mathbf{x}_k, \tilde{\mathbf{f}}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k))$  when the saturated state of the system  $\mathbf{x}_{k+1} \in \partial C(\mathbf{x}_k, \mathbf{u}_k)$  is obtained by projecting the unsaturated state of the system  $\tilde{\mathbf{f}}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k)$  onto the saturation boundary  $\partial C(\mathbf{x}_k, \mathbf{u}_k)$ .

A comparison of the PDF of an unsaturated variable  $\tilde{\mathbf{f}}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k)$  with the PDF of a saturated variable  $\mathbf{x}_{k+1}$  is presented in Figure 5.8.

Having the CSSDS defined in such a way, we are interested in estimating the actual state  $\mathbf{x}_k$  of the system from the available measurements  $\mathbf{y}_k$ . The Markovian character of the CSSDS makes it possible, for estimation purposes, to employ recursive algorithms utilizing Bayes' theorem, e.g., the PF.

There are many variations of PFs [Arulampalam et al., 2002], which employ various importance densities and resampling algorithms. For the sake of comparison, as a benchmark solution to the estimation problem suitable to saturated systems we chose the CBPF, i.e., the BPF [Arulampalam et al., 2002] modified by the projection approach of [Shao et al., 2010].

### 5.3.2 Convex Saturated Particle Filter

In this section we propose a new SPF that is designed for CSSDSs. The CSPF is capable of quickly detecting whether or not saturation occurred by comparing the measurements with the state constraints. This information is used to forcibly move the particles to the region of higher probability, which leads to improved accuracy of the estimate. This procedure renders possible the reduction of the number of particles used by PF, thus reducing the computational load of the algorithm. The detection of the saturation is achieved by a detection function introduced in Definition 5.6.

**Definition 5.6** (Detection function). *A function  $\alpha : \mathcal{Y} \times \mathcal{X} \times \Sigma_{\mathcal{X}} \rightarrow \mathbb{R}$  is called a detection function for a CSSDS if for every time step  $k$ , every precompact open convex set  $A \in \Sigma_{\mathcal{X}}$ ,*

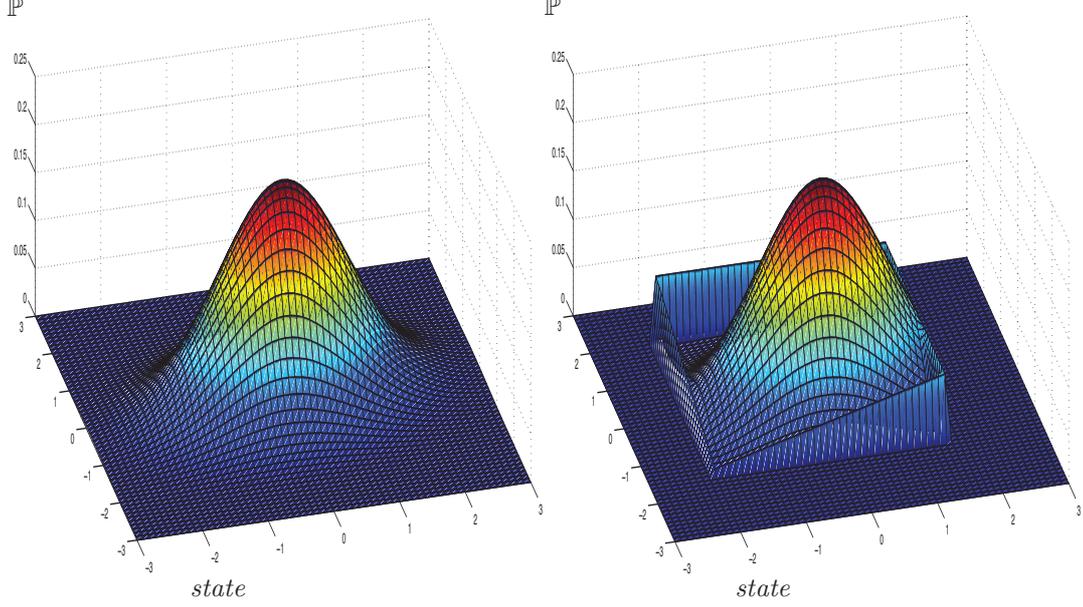


Figure 5.8: The PDF of the unsaturated variable  $\tilde{\mathbf{f}}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k)$  (left) and the PDF of the saturated variable  $\mathbf{x}_{k+1}$  (right) given the previous state  $\mathbf{x}_k$  and input  $\mathbf{u}_k$ . The PDF of the unsaturated variable is continuous (left) whereas the PDF of the saturated variable (right) has a continuous part (5.26a) and a singular mass (5.26b) concentrated on the saturation region  $\partial C(\mathbf{x}_k, \mathbf{u}_k) = \partial([-2, 1.5] \times [-1.5, 2])$ .

every  $\mathbf{y} \in \mathcal{Y}$  and  $\mathbf{x}_1, \mathbf{x}_2 \in \partial A$  the implication holds:

$$\left[ \|\mathbf{y} - \mathbf{h}_k(\mathbf{x}_1)\| \geq \|\mathbf{y} - \mathbf{h}_k(\mathbf{x}_2)\| \right] \implies [\alpha(\mathbf{y}, \mathbf{x}_1, \mathbf{h}_k(A)) \leq \alpha(\mathbf{y}, \mathbf{x}_2, \mathbf{h}_k(A))]. \quad (5.28)$$

Intuitively, a function for which the condition (5.28) holds serves as a ‘pseudo’-metric between the points at the boundary  $\partial A$  of the convex set  $A$  and, through the observation model  $\mathbf{h}_k$ , the measurement  $\mathbf{y} \in \mathcal{Y}$ . The properties of the detection functions are further explained while discussing the numerical example in Section 5.3.3.

Let us now consider the SSDS defined by (5.23). Furthermore, let  $\{(\mathbf{x}_k^i, \omega_k^i)\}_{i=1}^N$  be the approximation of the true PDF of the state at time step  $k$ . For each  $i \in \{1, \dots, N\}$ , given the previous particle  $\mathbf{x}_k^i$ , the probability that the particle  $\mathbf{x}_{k+1}^i$  will saturate, i.e.,  $\mathbf{x}_{k+1}^i \in \partial C(\mathbf{x}_k^i, \mathbf{u}_k)$  follows from (5.26b):

$$\mathbb{P}\left(\mathbf{x}_{k+1}^i \in \partial C(\mathbf{x}_k^i, \mathbf{u}_k)\right) = \int_{\partial C(\mathbf{x}_k^i, \mathbf{u}_k)} \left[ \int_{\mathcal{R}^1(\mathbf{x}_k^i, \mathbf{x})} \mathbb{P}\left(\tilde{\mathbf{f}}_k(\mathbf{x}_k^i, \mathbf{u}_k, \mathbf{w}_k) = \mathbf{z} | \mathbf{x}_k^i, \mathbf{u}_k\right) dH_1(\mathbf{z}) \right] dH_{n-1}(d\mathbf{x}), \quad (5.29)$$

where  $\mathcal{R}^1$  is defined by (5.27).

For the ease of notation the right-hand side of (5.29) is called the *predicted probability of saturation* and denoted as  $q_i$ , i.e.,

$$q_i := \int_{\partial C(\mathbf{x}_k^i, \mathbf{u}_k)} q_i(\mathbf{x}) dH_{n-1}(\mathbf{x}), \quad (5.30a)$$

## 5. SATURATED PARTICLE FILTER

---

where

$$q_i(\mathbf{x}) := \int_{\mathcal{R}^1(\mathbf{x}_k^i, \mathbf{x})} \mathbb{P}\left(\tilde{\mathbf{f}}_k(\mathbf{x}_k^i, \mathbf{u}_k, \mathbf{w}_k) = \mathbf{z} | \mathbf{x}_k^i, \mathbf{u}_k\right) dH_1(\mathbf{z}). \quad (5.30b)$$

The saturation of a particle  $\mathbf{x}_{k+1}^i$  can be seen as projecting  $\mathbf{x}_{k+1}^i$  onto  $\partial C(\mathbf{x}_k^i, \mathbf{u}_k)$  which is equivalent to the generic projection approach of [Shao et al., 2010]. Indeed, it makes no difference whether the ‘bad’ particles drawn from an unconstrained continuous distribution are projected on the saturation region, or each particle is set to the saturation region with the predicted probability of saturation. The resulting sets of particles are equivalent in the statistical sense.

Let  $\alpha$  be a given detection function satisfying Definition 5.6. Furthermore, assume that the measurement  $\mathbf{y}_{k+1}$  becomes available. Then, for each  $i \in \{1, \dots, N\}$  and each  $\mathbf{x} \in \partial C(\mathbf{x}_k^i, \mathbf{u}_k)$  we define the *updated probability of saturation*  $q_i^\alpha$ :

$$q_i^\alpha := \int_{\partial C(\mathbf{x}_k^i, \mathbf{u}_k)} q_i^\alpha(\mathbf{x}) dH_{n-1}(\mathbf{x}), \quad (5.31a)$$

where

$$q_i^\alpha(\mathbf{x}) \propto q_i(\mathbf{x}) + \alpha(\mathbf{y}_{k+1}, \mathbf{x}, C(\mathbf{x}_k^i, \mathbf{u}_k)). \quad (5.31b)$$

Using (5.30)–(5.31), and the detection function  $\alpha$ , we define the importance density  $\mathbb{Q}^\alpha$  of the new CSPF by:

$$\mathbb{Q}^\alpha(\mathbf{x} | \mathbf{x}_k^i, \mathbf{u}_k, \mathbf{y}_{k+1}) := q_i^\alpha(\mathbf{x}) \mathbf{1}_{\partial C(\mathbf{x}_k^i, \mathbf{u}_k)}(\mathbf{x}) \quad (5.32a)$$

$$+ \frac{1 - q_i^\alpha}{1 - q_i} \mathbb{P}\left(\tilde{\mathbf{f}}_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k) = \mathbf{x} | \mathbf{x}_k^i, \mathbf{u}_k\right) \mathbf{1}_{C(\mathbf{x}_k^i, \mathbf{u}_k)}(\mathbf{x}). \quad (5.32b)$$

It can be easily seen that  $\mathbb{Q}^\alpha$  defines a probability measure, i.e.,  $\mathbb{Q}^\alpha$  is positive, and it integrates to one. The importance density of the CBPF filter is a special case of  $\mathbb{Q}^\alpha$  with  $\alpha \equiv 0$ .

Given the particle  $\mathbf{x}_k^i$  and the input  $\mathbf{u}_k$ , a new particle  $\mathbf{x}_{k+1}^i$  is drawn from the importance density  $\mathbb{Q}^\alpha$ . The random sample from  $\mathbb{Q}^\alpha$  is obtained in two steps. First, the algorithm determines whether the particle  $\mathbf{x}_{k+1}^i$  saturates, i.e.,  $\mathbf{x}_{k+1}^i \in \partial C(\mathbf{x}_k^i, \mathbf{u}_k)$  (with the probability  $q_i^\alpha$ ) or not (with probability  $1 - q_i^\alpha$ ). Next, if saturation was detected, the particle  $\mathbf{x}_{k+1}^i$  is drawn from:

$$\mathbf{x}_{k+1}^i \sim \frac{q_i^\alpha(\cdot)}{q_i^\alpha} \mathbf{1}_{\partial C(\mathbf{x}_k^i, \mathbf{u}_k)}(\cdot). \quad (5.33)$$

In case the saturation was not detected the particle  $\mathbf{x}_{k+1}^i$  is drawn from:

$$\mathbf{x}_{k+1}^i \sim \frac{1}{1 - q_i} \mathbb{P}\left(\tilde{\mathbf{f}}_k(\mathbf{x}_k, \mathbf{w}_k) = \bullet | \mathbf{x}_k^i\right) \mathbf{1}_{C(\mathbf{x}_k^i, \mathbf{u}_k)}(\cdot). \quad (5.34)$$

The associated weights  $\omega_{k+1}^i$  are derived from the general principle (3.84) applied to the importance density (5.32). If  $\mathbf{x}_{k+1}^i$  saturates, then, by (5.30) and (5.31), the weight  $\omega_{k+1}^i$  is given by:

$$\omega_{k+1}^i \propto \omega_k^i \frac{q_i(\mathbf{x}_{k+1}^i)}{q_i^\alpha(\mathbf{x}_{k+1}^i)} \mathbb{P}\left(\mathbf{h}_{k+1}(\mathbf{x}_{k+1}, \mathbf{v}_k) = \mathbf{y}_{k+1} | \mathbf{x}_{k+1}^i\right), \quad (5.35)$$

if  $\mathbf{x}_{k+1}^i$  does not saturate, the weight  $\omega_{k+1}^i$  is updated by:

$$\omega_{k+1}^i \propto \omega_k^i \frac{1 - q_i}{1 - q_i^\alpha} \mathbb{P}\left(\mathbf{h}_{k+1}(\mathbf{x}_{k+1}, \mathbf{v}_k) = \mathbf{y}_{k+1} | \mathbf{x}_{k+1}^i\right). \quad (5.36)$$

---

**Algorithm 5.2** Convex Saturated Particle Filter
 

---

**Input:**  $\{(\mathbf{x}_k^i, \omega_k^i)\}_{i=1}^N, \mathbf{y}_{k+1}$ 
**Output:**  $\{(\mathbf{x}_{k+1}^i, \omega_{k+1}^i)\}_{i=1}^N$ 
**for**  $i = 1, 2, \dots, N$  **do**

 Compute the probability  $q_i$  according to (5.30a)

 Compute the probability  $q_i^\alpha$  according to (5.31a)

 Draw from the standard uniform distribution  $u \sim \mathcal{U}(0, 1)$ 
**if**  $u \leq q_i^\alpha$  **then**

 Particle  $\mathbf{x}_{k+1}^i$  saturates:

$$\begin{aligned} \mathbf{x}_{k+1}^i &\sim \frac{q_i^\alpha(\cdot)}{q_i^\alpha} \mathbf{1}_{\partial C(\mathbf{x}_k^i, \mathbf{u}_k)}(\cdot) \\ \omega_{k+1}^i &\propto \omega_k^i \frac{q_i(\mathbf{x}_{k+1}^i)}{q_i^\alpha(\mathbf{x}_{k+1}^i)} \mathbb{P}(\mathbf{h}_{k+1}(\mathbf{x}_{k+1}^i, \mathbf{v}_k) = \mathbf{y}_{k+1} | \mathbf{x}_{k+1}^i) \end{aligned}$$

**else**

 Particle  $\mathbf{x}_{k+1}^i$  does not saturate:

$$\begin{aligned} \mathbf{x}_{k+1}^i &\sim \frac{1}{1 - q_i} \mathbb{P}(\tilde{\mathbf{f}}_k(\mathbf{x}_k, \mathbf{w}_k) = \bullet | \mathbf{x}_k^i) \mathbf{1}_{C(\mathbf{x}_k^i, \mathbf{u}_k) \setminus \partial C(\mathbf{x}_k^i, \mathbf{u}_k)}(\cdot) \\ \omega_{k+1}^i &\propto \omega_k^i \frac{1 - q_i}{1 - q_i^\alpha} \mathbb{P}(\mathbf{h}_{k+1}(\mathbf{x}_{k+1}^i, \mathbf{v}_k) = \mathbf{y}_{k+1} | \mathbf{x}_{k+1}^i) \end{aligned}$$

**end if**
**end for**


---

The new CSPF is summarized in Algorithm 5.2.

The proposed CSPF combines the previous state  $\mathbf{x}_k^i$  with the most recent measurement  $\mathbf{y}_{k+1}$  to compute the updated probabilities of saturation  $q_i^\alpha(\mathbf{x})$  and the updated probability of saturation  $q_i^\alpha$ . The function

$$\partial C(\mathbf{x}_k^i, \mathbf{u}_k) \ni \mathbf{x} \mapsto \frac{q_i^\alpha(\mathbf{x})}{q_i^\alpha}, \quad (5.37)$$

is a continuous PDF on  $\partial C(\mathbf{x}_k^i, \mathbf{u}_k)$ .

Note that since each particle  $\mathbf{x}_k^i$  has dynamics of its own, for  $i \neq j$  the constraint regions  $C(\mathbf{x}_k^i, \mathbf{u}_k)$  and  $C(\mathbf{x}_k^j, \mathbf{u}_k)$  do not need to overlap. As a consequence it is possible that the  $i$ -th saturation region  $\partial C(\mathbf{x}_k^i, \mathbf{u}_k)$  nontrivially intersects with the admissible (unsaturated) part of the  $j$ -th constraint region  $C(\mathbf{x}_k^j, \mathbf{u}_k)$ . Thus, a region of the state space that is admissible to all the particles  $\{\mathbf{x}_k^i\}_i$  is given by  $\bigcap_i C(\mathbf{x}_k^i, \mathbf{u}_k)$ , and is also convex.

For large values of  $q_i^\alpha$  the algorithm forces the particles to be close to the saturation region  $\partial C(\mathbf{x}_k^i, \mathbf{u}_k)$  associated to the  $i$ -th particle, whereas for small values of  $q_i^\alpha$  the particles are set farther from the saturation region. Moreover, the particles that hit the boundary of the constraint region, thanks to (5.37), are forcibly moved to that part of the saturation region that has the highest probability. Figure 5.9 schematically describes the differences between the UBPf, the CBPF, and the CSPF for a large value of  $q_i^\alpha$ .

Finally, we have to mention that the accuracy of the CSPF estimate depends on the user-specified detection function, which must be chosen appropriately to the CSSDS under consid-

## 5. SATURATED PARTICLE FILTER

---

eration.

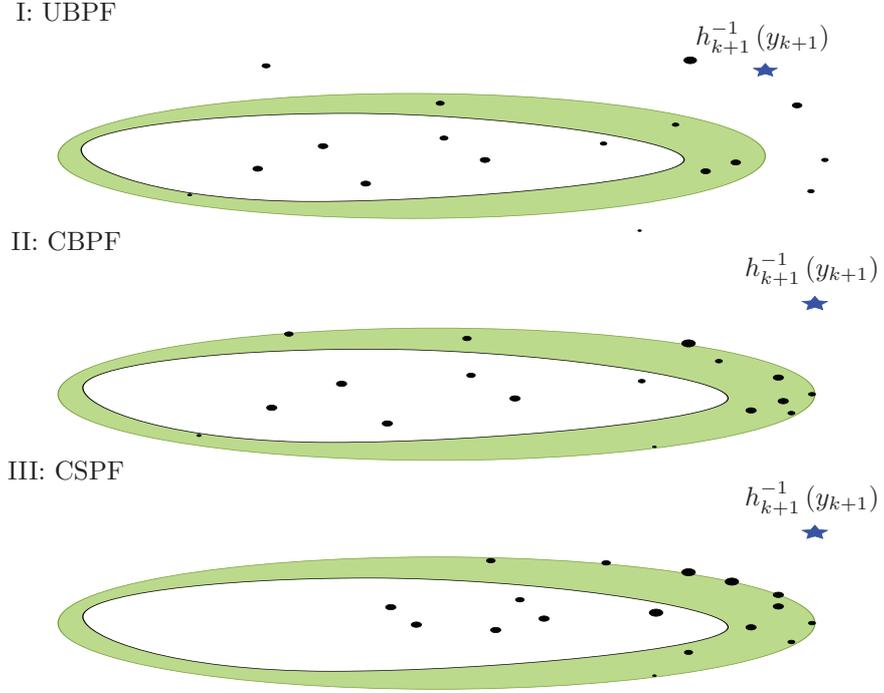


Figure 5.9: Visualization of the distribution of the particles obtained by the UBPF (top), the CBPF (middle) and the CSPF (bottom). Some of the particles obtained by the UBPF violate the physical constraints  $\bigcup_i C(\mathbf{x}_k^i, \mathbf{u}_k)$  of the system (shaded area and the region  $\bigcap C(\mathbf{x}_k^i, \mathbf{u}_k)$  bounded by it), others are located far from the actual measurement (star). The larger the size of a particle the higher its weight. The CBPF projects the unphysical particles onto the relevant saturation regions (inside the shaded area), but does not move the remaining particles. The CSPF projects the ‘bad’ particles onto the saturation region and forces the remaining particles to concentrate closer to the saturation region. Moreover, weights of the repositioned particles are appropriately rescaled.

### 5.3.3 Numerical Simulations

To illustrate the estimation abilities of the newly proposed CSPF we compare it with the CBPF applied to a simple CSSDS that models the motion of a two-dimensional object under random disturbance. We assume a static sensor placed at the origin that measures the distance and the bearing of the moving object. This model is a version of a classical nonlinear tracking problem discussed, e.g., in [Arulampalam et al., 2004; Gilks and Berzuini, 2001; Gordon et al., 1993]. In this motivating example we discuss in detail how to overcome the difficulties of practical implementations of the CSPF that arise from the extra integrations steps (5.30)–(5.31).

We consider the unconstrained system defined by:

$$\begin{bmatrix} x(1) \\ x(2) \end{bmatrix}_{k+1} = \begin{bmatrix} x(1) \\ x(2) \end{bmatrix}_k + T_s \begin{bmatrix} u(1) \\ u(2) \end{bmatrix}_k + w_k, \quad (5.38a)$$

$$\begin{bmatrix} y(1) \\ y(2) \end{bmatrix}_{k+1} = \begin{bmatrix} \sqrt{(x(1))^2 + (x(2))^2} \\ \arctan \frac{x(2)}{x(1)} \end{bmatrix}_k + v_k \quad (5.38b)$$

where  $\mathbf{w}_k$  and  $\mathbf{v}_k$  are two-dimensional zero-mean Gaussian variables with the covariance matrices  $\Sigma_x = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}$  and  $\Sigma_y = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.0012 \end{bmatrix}$ , respectively. The controlled deterministic input  $u$  corresponds to the velocity of the object and  $T_s = 1$  is the sampling period. The constraint imposed on such a system is defined by

$$C(\mathbf{x}_k, \mathbf{u}_k) = \left\{ \mathbf{x} : (\mathbf{x}_k + \mathbf{u}_k - \mathbf{x})^T (\mathbf{x}_k + \mathbf{u}_k - \mathbf{x}) \leq r^2 \|\mathbf{u}_k\|^2 \right\}, \quad (5.39)$$

where  $r$  is the user-specified parameter that, in our simulation, is set to  $r = 2$ . Thus, the constraint region  $C(\mathbf{x}_k, \mathbf{u}_k)$  is a ball centered in  $\mathbf{x}_k + \mathbf{u}_k$  with radius  $r\|\mathbf{u}_k\|$ .

The detection function  $\alpha$  for the CSPF measures the difference between the distances between the inverse measurement  $h_{k+1}^{-1}(\mathbf{y}_{k+1})$  (where  $h_k$  denotes the standard polar transformation) and the particle  $\mathbf{x}_k^i + \mathbf{u}_k$  and the vector at the boundary of the constraint region  $C(\mathbf{x}_k^i, \mathbf{u}_k)$ :

$$\alpha(h_{k+1}^{-1}(\mathbf{y}_{k+1}), \mathbf{x}, C(\mathbf{x}_k^i, \mathbf{u}_k)) = \theta \left( \|h_{k+1}^{-1}(\mathbf{y}_{k+1}) - \mathbf{x}_k^i - \mathbf{u}_k\| \right. \\ \left. \left( \|h_{k+1}^{-1}(\mathbf{y}_{k+1}) - \mathbf{x}_k^i - \mathbf{u}_k\| - \|h_{k+1}^{-1}(\mathbf{y}_{k+1}) - \mathbf{x}\| \right) \right), \quad (5.40)$$

for  $\mathbf{x} \in \partial C(\mathbf{x}_k^i, \mathbf{u}_k)$ , and user-specified parameter  $\theta$ . The strength of the influence of the detection function (5.40) depends on the value of the parameter  $\theta$ . We can see that the function defined by (5.40) satisfies the condition (5.28). In an extreme situation, when  $\theta = 0$ , the function  $\alpha$  has zero influence on the algorithm and the CSPF degenerates into the CBPF.

With the help of the function defined by (5.40) it is now possible to update the probability of saturation of all the particles  $\mathbf{x}_{k+1}^i$  by comparing the distances between the inverse measurement  $h_{k+1}^{-1}(\mathbf{y}_{k+1})$  (note that by (5.38b) the measurement space  $\mathcal{Y}$  and the state space  $\mathcal{X}$  are identical) and the points at the boundary of the constraint region  $C(\mathbf{x}_k^i, \mathbf{u}_k)$ .

To show how such an update can be achieved we need to consider two situations

- I. If the inverse observation  $h_{k+1}^{-1}(\mathbf{y}_{k+1})$  belongs to  $C(\mathbf{x}_k^i, \mathbf{u}_k)$ , then for all the points at the boundary  $\mathbf{x} \in \partial C(\mathbf{x}_k^i, \mathbf{u}_k)$  the updated probability of saturation  $q_i^\alpha$ 
  - increases, if  $\|\mathbf{x} - h_{k+1}^{-1}(\mathbf{y}_{k+1})\| \leq \|h_{k+1}^{-1}(h_{k+1}^{-1}(\mathbf{y}_{k+1})) - \mathbf{x}_k^i - \mathbf{u}_k\|$ ,
  - decreases, if  $\|\mathbf{x} - h_{k+1}^{-1}(h_{k+1}^{-1}(\mathbf{y}_{k+1}))\| > \|h_{k+1}^{-1}(\mathbf{y}_{k+1}) - \mathbf{x}_k^i - \mathbf{u}_k\|$ .
- II. For the inverse observation  $h_{k+1}^{-1}(\mathbf{y}_{k+1})$  that belongs to  $\mathcal{X} \setminus C(\mathbf{x}_k^i, \mathbf{u}_k)$  the updated probability of saturation  $q_i^\alpha$ 
  - increases, if  $\|\mathbf{x} - h_{k+1}^{-1}(\mathbf{y}_{k+1})\| \leq r\|\mathbf{u}_k\| + \text{dist}(h_{k+1}^{-1}(\mathbf{y}_{k+1}), C(\mathbf{x}_k^i, \mathbf{u}_k))$ ,
  - decreases, if  $\|\mathbf{x} - h_{k+1}^{-1}(\mathbf{y}_{k+1})\| > r\|\mathbf{u}_k\| + \text{dist}(h_{k+1}^{-1}(\mathbf{y}_{k+1}), C(\mathbf{x}_k^i, \mathbf{u}_k))$ .

To help understand the properties of the probabilities of saturation, the second of the discussed cases is illustrated in Figure 5.10.

In order to successfully implement the CSPF for the model (5.38a)–(5.39) we need to tackle the following technical problems:

## 5. SATURATED PARTICLE FILTER

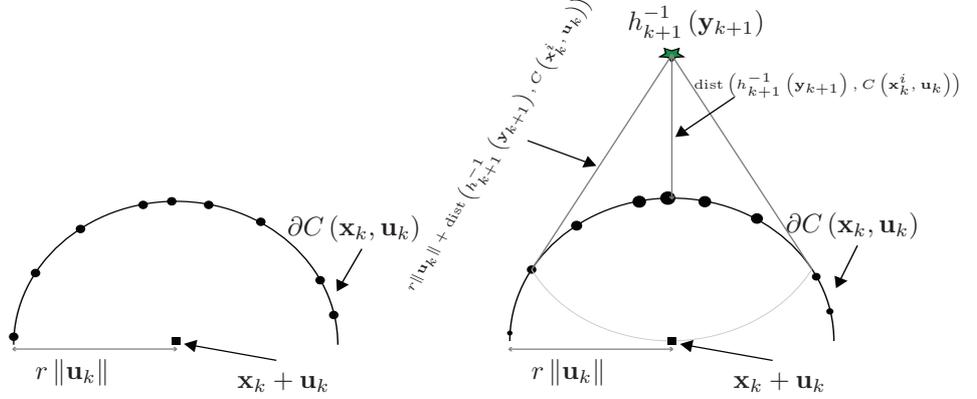


Figure 5.10: Visualization of the updated probability of saturation obtained with the use of the detection function (5.40). Before the measurement  $\mathbf{y}_{k+1}$  is collected (left) the probability of saturation is uniformly distributed on the boundary  $\partial C(\mathbf{x}_k, \mathbf{u}_k)$ . After the measurement  $\mathbf{y}_{k+1}$  becomes available (right) the probability of saturation is rescaled to account for the newest information.

- I. numerically evaluate the integrals (5.30),
- II. numerically evaluate the integral (5.31),
- III. draw random samples from the distribution  $\frac{q_i^\alpha(\cdot)}{q_i^\alpha}$ .

The first problem is solved in Proposition 5.1. The points 2 and 3 are closely connected hence are dealt with together.

**Proposition 5.1.** *The predicted probability of saturation  $q_i(\cdot)$  for the system defined by (5.38a)–(5.39) is uniformly distributed on the circle  $\partial C(\mathbf{x}_k^i, \mathbf{u}_k)$  and it integrates to*

$$q_i = 1 - \frac{1}{\Sigma_{11}} \int_0^{r\|\mathbf{u}_k\|} \exp\left(-\frac{z^2}{2\Sigma_{11}}\right) dz, \quad (5.41)$$

where  $\Sigma_{11}$  denotes the first diagonal entry of the matrix  $\Sigma_x$ .

*Proof.* The uniform distribution of  $q_i(\cdot)$  follows directly from the fact that  $\partial C(\mathbf{x}_k^i, \mathbf{u}_k)$  coincides with the  $\sigma$ -contours of the Gaussian variable  $w_k$ .

Formula (5.41) can be easily obtained by the polar parametrization of the plane  $\mathbb{R}^2$  and by observing that

$$\mathbb{P}(\mathbf{x}_{k+1} \in \partial C(\mathbf{x}_k, \mathbf{u}_k)) = 1 - \mathbb{P}(\mathbf{x}_{k+1} \in C(\mathbf{x}_k, \mathbf{u}_k)). \quad (5.42)$$

The probability in (5.42) is easily computed by:

$$\begin{aligned} & \mathbb{P}(\mathbf{x}_{k+1} \in C(\mathbf{x}_k, \mathbf{u}_k)) \\ &= \frac{1}{2\pi|\Sigma_x|^{1/2}} \int_{C(\mathbf{x}_k, \mathbf{u}_k)} \exp\left(-\frac{1}{2}\mathbf{x}^T \Sigma_x^{-1} \mathbf{x}\right) dx \end{aligned} \quad (5.43a)$$

$$= \frac{1}{2\pi\Sigma_{11}} \int_0^{2\pi} \int_0^{r\|\mathbf{u}_k\|} \exp\left(-\frac{z^2}{2\Sigma_{11}}\right) dz s\phi \quad (5.43b)$$

$$= \frac{1}{\Sigma_{11}} \int_0^{r\|\mathbf{u}_k\|} \exp\left(-\frac{z^2}{2\Sigma_{11}}\right) dz. \quad (5.43c)$$

Combining (5.42) with (5.43c) gives the desired (5.41).  $\square$

To compute (5.30) it is convenient to introduce the following parametrization of the boundary of the constraint region:

$$\partial C(\mathbf{x}_k, \mathbf{u}_k) = \{\mathbf{x}_k + \mathbf{u}_k + r\|\mathbf{u}_k\|(\cos(\phi), \sin(\phi)) : \phi \in [0, 2\pi)\}. \quad (5.44)$$

Then the updated probability of saturation (5.31b) can be seen as a function defined on the interval  $[0, 2\pi)$ :

$$[0, 2\pi) \ni \phi \mapsto q_i^\alpha(\mathbf{x}(\phi)), \quad (5.45)$$

hence the integral (5.30) can be efficiently evaluated, e.g., with the MATLAB function *quad.m*.

The parametrization (5.44) is also useful for sampling from  $\frac{q_i^\alpha(\cdot)}{q_i^\alpha}$  using the inverse of a *Cumulative Density Function* (CDF). The procedure of obtaining such a ‘pseudo’-random sample is described in Algorithm 5.3.

---

**Algorithm 5.3** Drawing from  $\frac{q_i^\alpha(\cdot)}{q_i^\alpha}$

---

Using (5.44) parametrize the boundary

$$\partial C(\mathbf{x}_k, \mathbf{u}_k) = \{\mathbf{x}(\phi) : \phi \in [0, 2\pi)\}$$

Compute the 1-dimensional Jacobian [Krantz and Parks, 2008]:

$$J(D_\varphi x) = \sqrt{\det(D_\varphi x^T D_\varphi x)} = rT_s\|\mathbf{u}_k\|$$

Compute the updated probability of saturation:

$$q_i^\alpha = \int_0^{2\pi} J(D_\varphi \mathbf{x}) q_i^\alpha(\mathbf{x}(\varphi)) dH_1(\varphi)$$

Define the CDF( $\phi$ ):

$$[0, 2\pi) \ni \phi \mapsto \int_0^\phi J(D_\varphi \mathbf{x}) \frac{q_i^\alpha(\mathbf{x}(\varphi))}{q_i^\alpha} d\varphi$$

Draw from the standard uniform distribution  $u \sim \mathcal{U}(0, 1)$

Find  $\phi_0$  such that  $\text{CDF}(\phi_0) = u$

Use  $\phi_0$  to obtain  $\mathbf{x}(\phi_0)$  - a random sample from  $\frac{q_i^\alpha(\cdot)}{q_i^\alpha}$

---

Note that  $\text{CDF}(\phi)$  is a continuous and monotone function that can be easily evaluated for any given  $\phi$ . Thus, a solution to  $\text{CDF}(\phi_0) = u$  can be obtained by applying any of the standard root-finding algorithms [Press et al., 2003].

Using the aforementioned numerical techniques, we have simulated the system (5.38) with the constraint (5.39) for  $T = 20$  time steps. In the simulations discussed in the remainder of the section the parameter in the detection function is set to  $\theta = \frac{0.005}{r\|\mathbf{u}_k\|}$  for appropriate scaling. The simulation started from the initial condition  $\mathbf{x}_0 = [10 \ 10]^T$  and the input signal  $\mathbf{u}_k = [3 \ 3]^T$  is constant over the whole simulation.

## 5. SATURATED PARTICLE FILTER

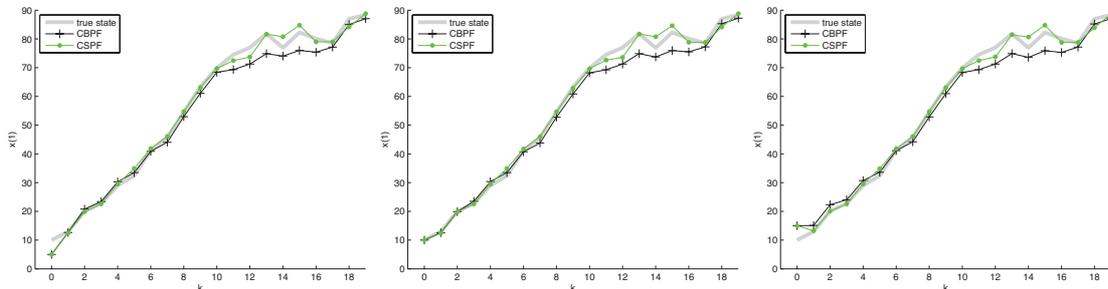


Figure 5.11: Tracking of the true state  $x_1$  (thick solid line) achieved by the CBPF (thin solid line with pluses) and by the CSPF (thin solid-dotted line). The estimates are obtained as the average of ten independent filters each utilizing 1000 particles. The estimates were obtained by the CBPF and the CSPF starting from initial positions:  $[5 \ 15]^T$  (left),  $[10 \ 10]^T$  (middle), and  $[15 \ 5]^T$  (right).

We tested the CBPF and the CSPF in three settings with different initial conditions. Namely, we simulated the case with no initial offset  $\mathbf{x}_0^{filters} = [10 \ 10]^T$ , and two scenarios with initial offsets:  $\mathbf{x}_0^{filters} = [5 \ 15]^T$  and  $\mathbf{x}_0^{filters} = [15 \ 5]^T$ . In each of the aforementioned settings the CBPF and the CSPF use the initial covariance  $\mathbf{P}_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ . Furthermore, for each scenario we test the CBPF and the CSPF with 10, 100, and 1000 particles.

To account for the probabilistic nature of the compared methods in every simulation setting we run the CBPF and the CSPF ten times, each utilizing the same observation sequence. The *average mean square error* (AMSE) of both filters obtained for all nine simulation scenarios are reported in Table 5.1. Table 5.2 reports the average times<sup>1</sup> required for the CBPF and the CSPF to produce a single-step estimate of the state.

Figures 5.11–5.13 report the averages of ten independent runs of the CBPF and the CSPF each using 1000 particles. The estimates were obtained by the CBPF and the CSPF starting from initial positions:  $[5 \ 15]^T$ ,  $[10 \ 10]^T$ , and  $[15 \ 5]^T$ . The outcomes of the estimation of the state variables  $x(1)$  and  $x(2)$  are presented in Figure 5.11 and Figure 5.12, respectively. The trajectory tracking is presented in Figure 5.13.

## 5.4 Conclusions and Discussion

Saturated Stochastic Dynamical Systems (SSDS) are severely nonlinear models that are often met in real life problems. Due to their complicated dynamical structure the states or the parameters of the SSDS can be accurately estimated only by non-parametric filters such as Particle Filters (PF).

In this chapter, a novel nonparametric filtering method has been derived. This method is designed specifically for SSDSs, i.e., stochastic systems with dynamics characterized by a constrained probability distribution exhibiting singularity on the boundary of the constraint region. The method hereby introduced exploits the specific structure of the SSDS in order to design an importance distribution that accounts for the most recent measurements in the

<sup>1</sup>The algorithm was executed in Matlab 7.9 on a Mac OS X with an Intel Core 2 Duo 2.66 GHz CPU with 4 GB RAM

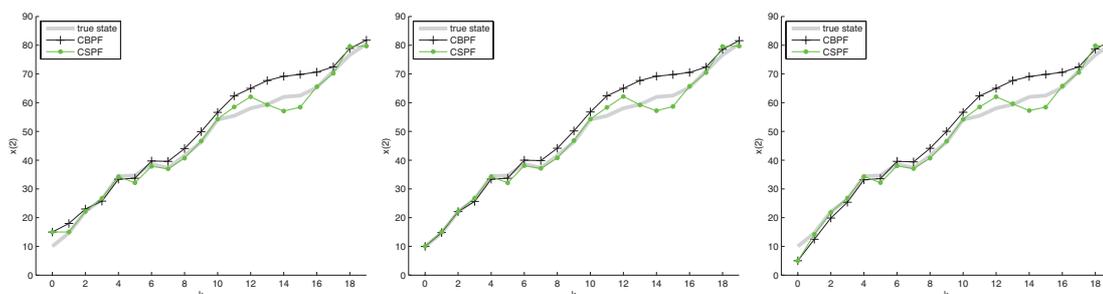


Figure 5.12: Tracking of the true state  $x_2$  (thick solid line) achieved by the CBPF (thin solid line with pluses) and by the CSPF (thin solid-dotted line). The estimates are obtained as the average of ten independent filters each utilizing 1000 particles. The estimates were obtained by the CBPF and the CSPF starting from initial positions:  $[5 \ 15]^T$  (left),  $[10 \ 10]^T$  (middle), and  $[15 \ 5]^T$  (right).

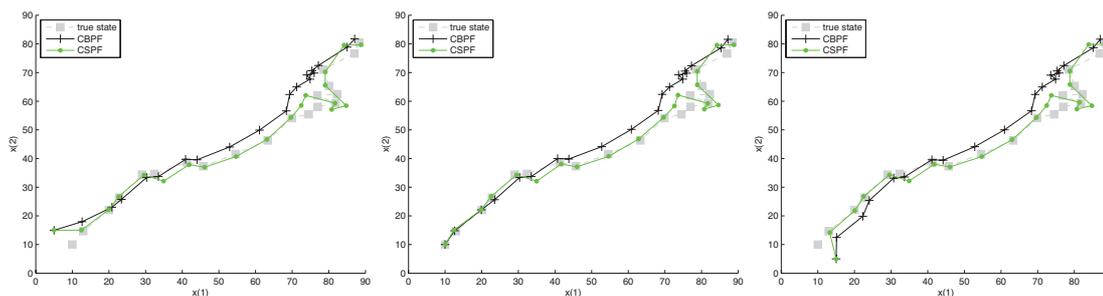


Figure 5.13: Trajectory of the true target (filled squares) and the estimates computed by the CBPF (solid line with pluses) and by the CSPF (solid-dotted line) in two dimensional plane. The estimates are obtained as the average of ten independent filters each utilizing 1000 particles. The estimates were obtained by the CBPF and the CSPF starting from initial positions:  $[5 \ 15]^T$  (left),  $[10 \ 10]^T$  (middle), and  $[15 \ 5]^T$  (right).

## 5. SATURATED PARTICLE FILTER

---

Table 5.1: AMSE of the CBPF and the CSPF

	Number of particles		
$x(0) = [15 \ 5]^T$	$N = 10^1$	$N = 10^2$	$N = 10^3$
CBPF	52.62	28.99	22.33
CSPF	9.03	6.69	5.91
	Number of particles		
$x(0) = [10 \ 10]^T$	$N = 10^1$	$N = 10^2$	$N = 10^3$
CBPF	53.78	23.61	20.05
CSPF	8.27	4.82	3.71
	Number of particles		
$x(0) = [5 \ 15]^T$	$N = 10^1$	$N = 10^2$	$N = 10^3$
CBPF	150.32	30.07	22.06
CSPF	12.41	6.72	5.88

Table 5.2: Average computational times (in seconds) of the CBPF and the CSPF

	Number of particles		
	$N = 10^1$	$N = 10^2$	$N = 10^3$
CBPF	0.0128	0.0365	0.2318
CSPF	0.1729	1.2692	12.7369

prediction step of the filtering algorithm. To achieve this we define a method of detecting the saturation of the particles from the collected measurements.

In Section 5.2 an efficient method for estimating the states of one-dimensional SSDSs, the Saturated Particle Filter (SPF), has been derived. The SPF combines the projection approach of [Shao et al., 2010] with a novel sampling method that effectively detects the saturation moment and forces the particles to rapidly jump to that part of the state space which is close to the saturation region. Such sampling is obtained by designing an importance density function that makes use of both the measurement and the knowledge of the system constraints.

The results of Section 5.3 extend the SPF introduced in Section 5.2.2, which makes an effective use of the measurements during the importance sampling, to the case of multidimensional SSDSs. Such extension requires an extra condition to be imposed on the system, namely the constraints of the system need to be convex sets in  $\mathbb{R}^n$ . With the convexity assumption satisfied, the multidimensional detection function can be properly defined. This function is then used to derive the multidimensional Convex Saturated Particle Filter (CSPF) that utilizes the measurements to detect the saturation of the system while sampling new particles.

The algorithms derived in Section 5.2 and in Section 5.3 focus on improving the importance sampling step of the PF methodology. Further improvements are possible by investigating the resampling procedure within the PF framework. This topic is out of the scope of this chapter and is instead discussed in details in Chapter 6.

The method has been tested in two case studies:

- I. The SPF has been applied to a one-dimensional SSDS whose dynamics depends on an unmodeled parameter that varies throughout the simulation. This models a shock change in the conditions external to the system and causes mismatch between the dynamics of the filter and the true dynamics of the system.
- II. The CSPF has been applied to a two-dimensional SSDS whose dynamics are influenced

by deterministic inputs.

In both simulation scenarios the performance of the proposed method has been compared to the Constrained Bootstrap Particle Filter (CBPF).

For the first case study the simulations of the noisy-measurement system demonstrated that the SPF outperforms the CBPF in terms of accuracy of tracking the signal that exhibit rapid changes in the dynamics. Furthermore, the results of the simulations suggest that the SPF is robust with respect to the choice of the detection function. While better performance is achieved, the computational complexity of the new filter is comparable to the complexity of the CBPF.

For the second case study the simulations showed that the CSPF outperforms the benchmark CBPF in terms of convergence speed as well as the accuracy of tracking the signal with saturated dynamics. The computational time required by the new filter is approximately 6.5 times larger than the computational time of the CBPF. However, to achieve good performance the CBPF requires many particles (more than 1000), whereas the CSPF is very accurate even when using few particles (10 particles). Thus, given that the computational complexity of the Particle Filter grows linearly with the number of particles used, the CSPF outperforms the benchmark CBPF in both accuracy and computational time.

Thus the results of both considered case studies showed that in each case our method outperforms the benchmark filter in terms of accuracy, speed of convergence, and numerical efficiency. The positive outcomes of the simulations call for the theory to back the results of the experimental studies. This call is answered in Chapter 6 where the asymptotic properties of the SPF are formally established.

Finally, we need to mention that, in general, the accuracy of the estimation depends on the appropriate choice of the detection function. The optimal construction of the detection function is still a matter of ongoing research.

## 5. SATURATED PARTICLE FILTER

---

## Chapter 6

# Asymptotic Properties of the Saturated Particle Filter

Parts of this chapter were published in:

- “*Saturated Particle Filter: Almost Sure Convergence and Improved Resampling*”, *Automatica*, 49 (1):1477-159, 2013, Paweł Mirosław Stano, Zsófia Lendek, and Robert Babuška.

### Abstract

Nonlinear stochastic dynamical systems are widely used to model physical processes. In many practical applications, the state variables are defined on a compact set of the state space, i.e., they are bounded or saturated. To estimate the states of systems with saturated variables, the Saturated Particle Filter (SPF) has been developed. This filter exploits the structure of the saturated system using a specific importance sampling distribution. In this chapter we investigate the asymptotic properties of the filter, in particular its almost sure convergence to the true posterior PDF. Furthermore, an improved SPF is developed that uses a novel resampling procedure to overcome the practical shortcomings of the original SPF. We prove that this new filter also converges almost surely to the true posterior PDF. Both versions of the SPF are presented in easy to implement algorithmic forms.

### 6.1 Introduction

As was indicated in Chapter 3, the *Particle Filter* (PF) approximates the true posterior PDF of the state of the dynamical system by a set of  $N$  discrete samples. Thus, the question that naturally arises is whether the approximation converges to the true posterior PDF as  $N \rightarrow \infty$ , and if yes in what sense? Extensive studies on the convergence properties of the PF have been conducted in [Cristian and Doucet, 2002; Doucet et al., 2001; Künsch, 2005]. In [Cristian and Doucet, 2002; Doucet et al., 2001] two types of convergence have been discussed, 1) almost sure convergence and 2) convergence in the mean square error sense, and conditions that guarantee either type of convergence have been derived. The focus of [Künsch, 2005] is more on investigating relations between the sample size  $N$  and the time step  $k$ . Moreover, [Künsch, 2005] presents a number of interesting results regarding asymptotic behavior of the variance of the estimator.

## 6. ASYMPTOTIC PROPERTIES OF THE SATURATED PARTICLE FILTER

In this chapter we consider the almost sure convergence of the *Saturated Particle Filter* (SPF) to the true posterior PDF. First, we derive sufficient conditions for the almost sure convergence of the SPF from Section 5.2.2. Next, we propose an improved version of the SPF with a novel resampling procedure and we prove that it also converges almost surely to the true posterior PDF. In both cases we discuss the practical meaning of the constraints that ensure the filter's convergence properties. Furthermore, both algorithms are presented in an easy-to-implement algorithmic form.

The chapter is structured as follows. In Section 6.2 we present background information regarding stochastic systems theory. In Section 6.3 we discuss asymptotic properties of the SPF described in Section 5.2.2 and originally proposed in [Stano et al., 2011]. The improved version of the SPF is derived in Section 6.4, where we also prove the almost sure convergence of the new *improved Saturated Particle Filter* (iSPF) to the true posterior PDF. Section 6.5 concerns practical considerations regarding the implementation of the iSPF and we discuss the practical properties of the detection function. Furthermore, we compare the iSPF and the SPF with the benchmark BPF in numerical simulations. Section 6.6 concludes the chapter.

### 6.2 Preliminaries

We start this section with giving the basic definitions that are further used to establish the asymptotic properties of the SPF. Then, we recall the theorems dealing with the convergence properties of the PFs. A comprehensive overview of the presented topics can be found in [Cristian and Doucet, 2002; Doucet et al., 2001; Meyn and Tweedie, 1993].

**Definition 6.1** (Feller kernel). *The transition probability kernel  $K(-|\cdot)$  on  $(\mathcal{X}, \Sigma_{\mathcal{X}}, \mathbb{P})$  has the Feller property<sup>1</sup> if, for every continuous and bounded function  $\varphi$ , the function*

$$z \rightarrow \int_{\mathcal{X}} \varphi(x) K(dx|z) \quad (6.1)$$

*is continuous and bounded [Doucet et al., 2001].*

In the following, we show that the PF can be defined as an operator on the space of probabilistic measures  $\mathcal{P}(\mathcal{X})$ , where  $\mathcal{X}$  is a given vector space. The construction of such an abstract operator requires some extra effort, but it allows us to derive simple conditions that guarantee good asymptotic behavior of a generic PF.

We start by introducing the operators [Cristian and Doucet, 2002] that will be used in the proof of convergence.

Let  $K_k$  be a transition probability kernel on the probability space  $(\mathcal{X}, \Sigma_{\mathcal{X}}, \mathbb{P})$  defined by (3.78) and let  $\tilde{K}_k$  be an arbitrary probability kernel that is absolutely continuous with respect to  $K_k$ . Furthermore, let  $g_k$  be a likelihood function defined by (3.79) and let  $\omega_k$  be a *weighted likelihood function* defined by:

$$\omega_k(y, x|x_k) := \frac{g_{k+1}(y|x) K_k(x|x_k)}{\tilde{K}_k(x|x_k, y)}. \quad (6.2)$$

**Definition 6.2** (Prediction operator). *The prediction operator  $b_k$  maps the probabilistic measure  $\nu \in \mathcal{P}(\mathcal{X})$  into the probabilistic measure  $b_k(\nu) \in \mathcal{P}(\mathcal{X})$ , defined by*

$$[b_k(\nu)](A) := \int_{\mathcal{X}} \tilde{K}_k(A|x_k, y_{k+1}) \nu(dx_k), \quad (6.3)$$

<sup>1</sup>Weak Feller property by the definition of [Meyn and Tweedie, 1993].

for every  $A \in \Sigma_{\mathcal{X}}$ .

**Definition 6.3** (Update operator). *For given  $y_{k+1}$  and  $x_k$ , the update operator  $a_k$  maps the probabilistic measure  $\nu \in \mathcal{P}(\mathcal{X})$  into the probabilistic measure  $a_k(\nu) \in \mathcal{P}(\mathcal{X})$ , defined by*

$$\int_{\mathcal{X}} \varphi(x) [a_k(\nu)](dx) := \left[ \int_{\mathcal{X}} \varphi(x) \omega_k(y_{k+1}, x|x_k) \nu(dx) \right] \left[ \int_{\mathcal{X}} \omega_k(y_{k+1}, x|x_k) \nu(dx) \right]^{-1} \quad (6.4)$$

for every continuous and bounded function  $\varphi$ .

**Definition 6.4** (Multinomial sampling operator). *The multinomial sampling operator  $c^N$  assigns to the probabilistic measure  $\nu \in \mathcal{P}(\mathcal{X})$  its random discrete approximation  $c^{N,x}(\nu)$  according to:*

$$c^{N,x}(\nu) := \frac{1}{N} \sum_{j=1}^N \delta_{\{V_j(x)\}}, \quad (6.5)$$

where  $N > 0$ ,  $x \in \mathcal{X}$ ,  $V_j$ ,  $j = 1, \dots, N$  are i.i.d. random variables on  $\mathcal{X}$  with the common distribution  $\nu$ .

Equation (6.5) formally defines the empirical approximation of an arbitrary distribution  $\nu$  by means of Monte Carlo sampling.

A special case of importance sampling, which is employed in the BPF, is when it is desirable to directly draw samples from the transition kernel  $K_k(-|\cdot)$ . In such a case  $\tilde{K}_k(-|\cdot) := K_k(-|\cdot)$ , and  $\omega_k$  reduces to  $g_k$ .

**Definition 6.5** (Particle Filter). *Let  $\bar{c}^N$  be the resampling operator on  $\mathcal{P}(\mathcal{X})$  that maps the measure  $\nu$  into a random measure  $\bar{c}^{N,\cdot} \in \mathcal{P}(\mathcal{X})$  composed of  $N$  discrete random measures. The Particle Filter is an operator  $k_k^N$  that transforms the empirical measure  $\pi_{k|k}^N$ , which approximates the state of the system at time  $k$ , into the empirical measure  $\pi_{k+1|k+1}^N$  at time  $k+1$ :*

$$\pi_{k+1|k+1}^N = k_k^N \left( \pi_{k|k}^N \right) := [\bar{c}^N \circ a_k \circ c^N \circ b_k] \left( \pi_{k|k}^N \right). \quad (6.6)$$

The relation between this abstract definition and the standard formulation of the PF is the following:

- I. *Prediction stage:* First, the predicted state density is computed by applying the operator  $b_k$ , defined in Definition 6.2, to the empirical measure  $\pi_{k|k}^N$ . Then the predicted state density is approximated by  $N$  random samples obtained by applying the sampling operator  $c^N$  (Definition 6.4) to  $b_k \left( \pi_{k|k}^N \right)$ .
- II. *Update stage:* After the prediction stage, the updated state density is computed as the output of the update operator  $a_k$  (Definition 6.3) applied to  $c^N \circ b_k \left( \pi_{k|k}^N \right)$ . Finally, applying the operator  $\bar{c}^N$  to the updated state density corresponds to the resampling step of the PF.

The asymptotic properties of the PF can be established as the following theorem:

## 6. ASYMPTOTIC PROPERTIES OF THE SATURATED PARTICLE FILTER

---

**Theorem 6.1** (Convergence of the generic PF [Cristian and Doucet, 2002]). *Let us assume that for each  $k$  the importance kernel  $\tilde{K}_k$  is Feller, and the likelihood function  $\omega_k$  is bounded, continuous, and strictly positive. Furthermore, let  $\bar{c}^N$  be a resampling operator such that for every bounded function  $\varphi$ , there exists a constant  $L$  such that:*

$$\mathbb{E} \left[ \left( \int_x \varphi(x) [\bar{c}^{N,\cdot}(\nu)](dx) - \int_x \varphi(x) \nu(dx) \right)^4 \right] \leq \frac{L}{N^2}. \quad (6.7)$$

*Then, as  $N \rightarrow \infty$ , the empirical measure  $\pi_{k+1|k+1}^N$  defined by (6.6) converges almost surely towards the true posterior PDF  $\pi_{k+1|k+1}$ .*

### 6.3 Asymptotic Properties of the SPF Under Standard Resampling

In this section we investigate asymptotic properties of the SPF with respect to Theorem 6.1. First, we prove the theoretical convergence of the SPF. Second, we discuss the practical consequences of the convergence conditions.

#### 6.3.1 Theoretical Results

We start with formulating the SPF algorithm in terms of the operator notation introduced in Section 6.2.

**Definition 6.6** (Saturated Particle Filter). *Consider the SSDS defined by (5.1a)-(5.1c) and (5.24), and let  $\alpha$  be an arbitrary detection function satisfying Definition 5.6. Furthermore, let  $K_k$  be the transition probability kernel and  $g_k$  be the likelihood function corresponding to the state model (5.1a) and the observation model (5.1b) respectively. The Saturated Particle Filter (SPF) is a PF with the transition probability kernel  $\tilde{K}_k$  defined by:*

$$\begin{aligned} \tilde{K}_k(x|x_k, y_{k+1}) &:= q_{k+1}^\alpha(x_k, y_{k+1}) \delta_{\{C(x_k)\}}(x) \\ &\quad + \frac{1 - q_{k+1}^\alpha(x_k, y_{k+1})}{1 - q_{k+1}^\alpha(x_k)} K_k(x|x_k) \mathbf{1}_{[0, C(x_k)]}(x), \end{aligned} \quad (6.8)$$

and with the weighted likelihood function defined by

$$\tilde{\omega}_k(y, x|x_{k-1}) := g_k(y|x) \left( \frac{1 - q_k(x_{k-1})}{1 - q_k^\alpha(x_{k-1}, y)} \mathbf{1}_{[0, C(x_{k-1})]}(x) + \frac{q_k(x_{k-1})}{q_k^\alpha(x_{k-1}, y)} \delta_{C(x_{k-1})}(x) \right), \quad (6.9)$$

where the predicted probability of saturation  $q_k$  and the updated probability of saturation  $q_k^\alpha$  are defined as:

$$q_k(x) := \int_{C(x)}^{+\infty} K_k(dz|x), \quad (6.10)$$

$$q_k^\alpha(x, y) := \begin{cases} 1 & \text{if } q_k(x) + \alpha(y - h_{k+1}(C(x))) > 1, \\ 0 & \text{if } q_k(x) + \alpha(y - h_{k+1}(C(x))) < 0, \\ q_k(x) + \alpha(y - h_{k+1}(C(x))) & \text{otherwise.} \end{cases} \quad (6.11)$$

### 6.3. Asymptotic Properties of the SPF Under Standard Resampling

It is easy to see that Definition 6.6 is an abstraction of Algorithm 5.2.

In what follows we derive sufficient conditions that, if satisfied, ensure the almost sure convergence of the SPF to the true posterior PDF  $\pi_{k+1|k+1}$ . We start with two technical lemmas:

**Lemma 6.1.** *Let  $K_k$  be a bounded Feller kernel and let  $C : \mathbb{R}_+ \rightarrow \mathbb{R}_+$  be a continuous function. Then, for every bounded and continuous function  $\varphi$ , the function*

$$z \rightarrow \int_0^{C(z)} \varphi(x) K_k(dx|z) \quad (6.12)$$

is continuous.

*Proof.* Let  $\{z_n\}$  be a sequence in  $\mathbb{R}_+$  such that  $z_n \rightarrow z_0$  as  $n \rightarrow +\infty$ . We have:

$$\left| \int_0^{C(z_n)} \varphi(x) K_k(dx|z_n) - \int_0^{C(z_0)} \varphi(x) K_k(dx|z_0) \right| \leq \left| \int_0^{C(z_n)} \varphi(x) K_k(dx|z_n) - \int_0^{C(z_0)} \varphi(x) K_k(dx|z_n) \right| \quad (6.13a)$$

$$+ \left| \int_0^{C(z_0)} \varphi(x) K_k(dx|z_n) - \int_0^{C(z_0)} \varphi(x) K_k(dx|z_0) \right|. \quad (6.13b)$$

The term (6.13b) converges to zero by the Feller property of  $K_k$ . Thus, let us focus on (6.13a). We have:

$$\left| \int_0^{C(z_n)} \varphi(x) K_k(dx|z_n) - \int_0^{C(z_0)} \varphi(x) K_k(dx|z_n) \right| = \left| \int_{C(z_0)}^{C(z_n)} \varphi(x) K_k(dx|z_n) \right| \quad (6.14a)$$

$$\leq \|\varphi\|_\infty \|K_k(\cdot|z_n)\|_\infty \left| \int_{C(z_0)}^{C(z_n)} dx \right| \quad (6.14b)$$

$$= \|\varphi\|_\infty \|K_k(\cdot|z_n)\|_\infty |C(z_n) - C(z_0)|. \quad (6.14c)$$

The term (6.14c) converges to zero by the continuity of  $C$ .  $\square$

**Lemma 6.2.** *Assume that  $K_k$  is a bounded Feller kernel. Furthermore, let  $\alpha$ ,  $C$ ,  $z \rightarrow K_k(z|\cdot)$  and  $h_{k+1}$  be continuous functions. Then, the kernel  $\tilde{K}_k$  defined by (6.8) has the Feller property.*

*Proof.* First, let us observe that by Lemma 6.1 the function  $q_k$  is continuous. Consequently, the function  $q_k^\alpha$  is continuous by the continuity of  $q_k$ ,  $\alpha$ ,  $C$  and  $h_{k+1}(\cdot, 0)$ .

Let  $\varphi$  be a bounded continuous function on  $\mathbb{R}$ . Then the following holds:

$$\int_{\mathbb{R}} \varphi(x) \tilde{K}_k(dx|z, y) = q_{k+1}^\alpha(z, y) \varphi(C(z)) \quad (6.15a)$$

$$+ \frac{1 - q_{k+1}^\alpha(z, y)}{1 - q_{k+1}(z)} \int_0^{C(z)} \varphi(x) K_k(dx|z). \quad (6.15b)$$

The continuity of the functions in (6.15b) follows by the continuity of  $q_k$ ,  $q_k^\alpha$ ,  $\alpha$ ,  $C$  and by Lemma 6.1.  $\square$

## 6. ASYMPTOTIC PROPERTIES OF THE SATURATED PARTICLE FILTER

---

**Lemma 6.3.** *Assume that the likelihood function  $g_k$  is bounded, continuous and strictly positive. Also, let  $\alpha, C, z \rightarrow K_k(z|\cdot)$  and  $h_{k+1}$  be continuous functions. Furthermore, let us assume that there exist positive constants  $M_1$  and  $M_2$  such that for every  $x \in \mathbb{R}$  it holds:*

$$0 < M_1 \leq q_k(x) \leq M_2 < 1. \quad (6.16)$$

*Then, if the detection function  $\alpha$  is chosen so that it satisfies the condition:*

$$\forall x \in \mathbb{R} : \quad -M_1 < \alpha(x) < 1 - M_2, \quad (6.17)$$

*the weighted likelihood function  $\omega_k$  defined by (6.9) is continuous, bounded and strictly positive.*

*Proof.* Continuity of  $\omega_k$  follows from the continuity of  $g_k, q_k, q_k^\alpha$ , and  $C$ .

By (6.17) there exists  $\epsilon > 0$  such that:

$$-M_1 + \epsilon \leq \alpha(x) \leq 1 - M_2 - \epsilon \quad (6.18)$$

holds for every  $x \in \mathbb{R}$ . Hence, for every  $x, y \in \mathbb{R}$  we have

$$q_k^\alpha(x, y) = q_k(x) + \alpha(y - h_{k+1}(C(x))) \quad (6.19a)$$

$$\leq M_2 + 1 - M_2 - \epsilon \quad (6.19b)$$

$$\leq 1 - \epsilon. \quad (6.19c)$$

Similarly, we deduce that for every  $x, y \in \mathbb{R}$  it holds:

$$q_k^\alpha(x, y) \geq \epsilon. \quad (6.20)$$

Therefore, by (6.9), we have

$$\|\omega_k\|_\infty \leq \|g_k\|_\infty \frac{1}{\epsilon}, \quad (6.21)$$

hence,  $\omega_k$  is bounded.

Finally, strict positivity of  $\omega_k$  follows by the strict positivity of  $g_k$  and by (6.16).  $\square$

The asymptotic properties of the SPF are described by the next theorem:

**Theorem 6.2** (Convergence of the SPF:  $c^N$ -resampling). *Let us consider a SPF  $k_k^N$  with the resampling operator  $c^N$  defined by Definition 6.4. If*

- I.  $K_k$  is a bounded Feller kernel,
- II.  $g_k$  is bounded, continuous and strictly positive,
- III.  $\alpha, C, z \rightarrow K_k(z|\cdot)$  and  $h_{k+1}$  are continuous functions,
- IV. conditions (6.16) and (6.17) are satisfied,

*then  $k_k^N \left( \pi_{k|k}^N \right)$  converges almost surely towards the true posterior PDF  $\pi_{k+1|k+1}$ .*

*Proof.* It has been proven by [Cristian and Doucet, 2002] that the multinomial sampling operator  $c^N$  satisfies (6.7). Furthermore, by Lemmas 6.1–6.3, the kernel  $\tilde{K}_k$  is Feller, and the weighted likelihood function  $\omega_k$  is bounded, continuous and strictly positive. Therefore, by Theorem 6.1,  $k_k^N \left( \pi_{k|k}^N \right)$  converges almost surely towards the true posterior PDF  $\pi_{k+1|k+1}$ .  $\square$

### 6.3.2 Practical Considerations

Let us now discuss the meaning of conditions I)–IV) of Theorem 6.2 from the practical perspective.

Assumptions I)–III) of Theorem 6.2 ensure that the model is “appropriately regular”, which is the case in most real life applications. Therefore, we can safely conclude that conditions I)–III) are not very restrictive from the practical point of view.

Assumption IV) of Theorem 6.2 is more problematic. In particular, ensuring that (6.16) is satisfied is not trivial, because often the function  $q_k$  cannot be evaluated analytically [Stano et al., 2010]. Fortunately, in practice, we do not need to compute the values  $q_k(x)$  for every  $x \in \mathbb{R}$ . It is sufficient to check whether (6.16) holds for every particle  $x_{k-1}^i$ , i.e., we need to check whether there exist positive constants  $M_1$  and  $M_2$  such that for every  $i = 1, \dots, N$

$$0 < M_1 \leq q_k^i \leq M_2 < 1 \quad (6.22)$$

holds.

Obviously, (6.22) is satisfied if and only if

$$\min_i \{q_k^i\} > 0, \quad (6.23a)$$

$$\max_i \{q_k^i\} < 1. \quad (6.23b)$$

Hence, for each particle the saturation event is possible, but not certain.

If the conditions in (6.23) are satisfied, we can choose  $\alpha$  such that

$$-\min_i \{q_k^i\} (1 - \epsilon) \leq \alpha \leq \left(1 - \max_i \{q_k^i\}\right) (1 - \epsilon), \quad (6.24)$$

where  $\epsilon > 0$  is small enough so that  $\alpha$  is nontrivial.

The advantage of choosing  $M_1$ ,  $M_2$  and  $\alpha$  so that (6.22) and (6.24) hold, is the low computational complexity of determining  $M_1$ ,  $M_2$ , and  $\alpha$  that satisfy condition IV) of Theorem 6.2. However, this approach has two shortcomings that need to be tackled.

- I.  $\alpha$  becomes recursive. By (6.24) we see that the conditions that  $\alpha$  needs to satisfy depend on the values of  $q^i$  at time step  $k$ . Thus,  $\alpha$  is not any more defined for all time steps  $k$ , but it becomes a recursive function  $\alpha_k$  that needs to be updated at each iteration of the algorithm.
- II.  $\alpha$  becomes negligible. Since both min and max are monotonic functions, with the increasing number of particles the image of  $\alpha_k$  becomes narrower. This means that the influence of  $\alpha_k$  becomes negligible, hence the SPF becomes undistinguishable from the BPF.

Both issues are addressed in the next section.

## 6.4 Asymptotic Behavior of the SPF Under Improved Resampling

In this section we derive an improved SPF algorithm that allows for a recursive computation of the detection function  $\alpha_k$ . Furthermore, by introducing a new resampling procedure, we make sure that at each time step  $k$  the influence of  $\alpha_k$  is not trivial.

### 6.4.1 Motivation

As it was indicated in Section 6.3, if there exists a particle  $\mathbf{x}^i$  such that  $q^i \approx 0$  or  $q^i \approx 1$ , then  $\alpha$  becomes approximately zero, and therefore its influence becomes negligible. If the weight  $\omega^i$  of such a particle  $\mathbf{x}^i$  is close to 1 it means that the uncertainty associated with the estimate is very small. Therefore, in the next filtering step, it is reasonable to “trust” the model and limit the influence of the noisy measurement on the subsequent estimate. In general, the same reasoning holds if there exists a small  $\epsilon$  such that in one of the intervals  $[0, \epsilon]$  or  $[1 - \epsilon, 1]$  there are enough  $q^i$ s so that the weights of the associated particles almost sum up to 1.

The situation is fundamentally different when we encounter a low weighted particle  $\mathbf{x}^i$  such that either  $q^i \approx 0$  or  $q^i \approx 1$ . The probability of such an event is very high, especially when we use a large set of particles, yet such a particle does not give us any important information about the system. Nevertheless, by (6.24), the existence of such a particle significantly decreases the influence of  $\alpha$ . To avoid this undesirable situation we need to discard the low weighted particles such that the corresponding  $q^i$ s lay in either of the intervals  $[0, \epsilon]$  or  $[1 - \epsilon, 1]$ , and in their place resample an equal number of particles in the “high probability” regions. Such resampling only slightly influences the posterior PDF  $\pi_{k|k}^N$  that approximates the true posterior PDF  $\pi_{k|k}$ . This is because the discussed resampling procedure cuts only the “light” tails, i.e., tails with negligible probability mass, thus also the resampled particles add insignificant weights to the approximation  $\pi_{k|k}^N$ . Nevertheless, by applying such resampling algorithm we are sure that  $\alpha$  is not trivial, and that the interval  $[-\epsilon, \epsilon]$  is in the image of  $\alpha$ , i.e.,  $[-\epsilon, \epsilon] \subset \alpha(\mathbb{R})$ . Furthermore, the number of particles remains constant throughout the filtering.

### 6.4.2 New Resampling

Let us now formalize the heuristic approach described in Section 6.4.1. Following the convention described in Section 5.2.2 we introduce a new resampling procedure by defining an operator  $\bar{c}^N$  acting on the space of probability measures on  $(\mathbb{R}_+, \Sigma_{\mathbb{R}_+})$ . To define such an operator we first need to introduce the concept of the  $\epsilon$ -set:

**Definition 6.7** ( $\epsilon$ -set). *Consider the SPF setting according to Definition 6.6, and let  $q_k$  be a function defined by (6.10). For a given  $\epsilon > 0$  we define the  $\epsilon$ -set  $\Omega_\epsilon$  by:*

$$\Omega_\epsilon := \{x \in \mathbb{R}_+ : 1 - \epsilon > q_k(x) > \epsilon\}. \quad (6.25)$$

Let us now consider an arbitrary probability measure  $\nu$  on  $(\mathbb{R}_+, \Sigma_{\mathbb{R}_+})$ . For a given  $\epsilon$  we define a new probabilistic measure  $\nu_\epsilon$  as a measure  $\nu$  conditioned on  $\Omega_\epsilon$ , i.e., for every  $A \in \Sigma_{\mathbb{R}_+}$  it holds:

$$\nu_\epsilon(A) := \nu(A|\Omega_\epsilon). \quad (6.26)$$

**Definition 6.8** (Resampling operator for the SPF). *Let  $\nu$  be a probabilistic measure on  $(\mathbb{R}_+, \Sigma_{\mathbb{R}_+})$ , and let  $\tilde{\epsilon} > 0$  be a given constant. Let  $\epsilon_0 > 0$  be the maximal positive constant such that the  $\nu$ -measure on  $\Omega_{\epsilon_0}$  is greater or equal to  $1 - \tilde{\epsilon}$ , i.e.,  $\epsilon_0$  is given by:*

$$\epsilon_0 := \max \{\epsilon : \nu(\Omega_\epsilon) \geq 1 - \tilde{\epsilon}\}. \quad (6.27)$$

*The new resampling operator  $\bar{c}^N$  assigns to every probabilistic measure  $\nu \in \mathcal{P}(\mathbb{R}_+, \Sigma_{\mathbb{R}_+})$  its random approximation  $\bar{c}^{N,x}(\nu)$  given by*

$$\bar{c}^{N,x}(\nu) := \frac{1}{N} \sum_{j=1}^N \delta_{\tilde{V}_j(x)}, \quad (6.28)$$

## 6.4. Asymptotic Behavior of the SPF Under Improved Resampling

---

where  $\{\tilde{V}_j\}_{j=1}^N$  is a set of  $N$  i.i.d. random variables distributed according to  $\nu_{\epsilon_0}$ .

Thus, the newly defined resampling operator  $\bar{c}^N$ , when applied to a measure  $\nu$ , returns a probabilistic measure concentrated on the set that has a  $\nu$ -measure close to  $1 - \tilde{\epsilon}$ . When used in the recursive framework of the SPF, the operator  $\bar{c}^N$  guarantees that the interval  $[-\epsilon_0, \epsilon_0]$  is in the image of the detection function  $\alpha$ , i.e.,  $[-\epsilon_0, \epsilon_0] \subset \alpha(\mathbb{R})$ .

Note that the operator  $\bar{c}^N$  depends on both the sample size  $N$  and on the constant  $\tilde{\epsilon} > 0$ . In what follows we show that the choice of  $\tilde{\epsilon}$  is not arbitrary but it is strictly determined by the sample size  $N$ . This is why in Definition 6.8 we did not use any symbol indicating the dependency of  $\bar{c}^N$  on  $\tilde{\epsilon}$ .

### 6.4.3 Almost Sure Convergence

In this section we prove that as the number of samples  $N$  increases, the SPF with the resampling operator  $\bar{c}^N$  from Definition 6.8 converges almost surely to the true posterior PDF  $\pi_{k+1|k+1}$ .

First, we prove the following two lemmas.

**Lemma 6.4.** *Let  $\nu$  be an arbitrary probabilistic measure on  $(\mathbb{R}_+, \Sigma_{\mathbb{R}_+})$  and let  $\tilde{\epsilon} > 0$  be a given positive constant. Furthermore, let  $\{\tilde{V}_j\}_{j=1}^N$  be a set of i.i.d. variables with a common distribution  $\nu_{\epsilon_0}$ , with  $\epsilon_0$  defined in (6.27). Finally, let  $\varphi$  be a continuous and bounded function on  $\mathbb{R}_+$ . Then, for every  $1 \leq j \leq N$  the following holds:*

$$\mathbb{E}_\nu \left( \varphi(\tilde{V}_j) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right) \leq 2\|\varphi\|_\infty \tilde{\epsilon} \quad (6.29)$$

where  $\|\cdot\|_\infty$  is the supremum norm on  $\mathbb{R}_+$ .

*Proof.* By (6.26) the distribution  $\nu_{\epsilon_0}$  of the variables  $\tilde{V}_j$  is a measure  $\nu$  conditioned on the set  $\Omega_{\epsilon_0}$ . Therefore, we have:

$$\left| \mathbb{E}_\nu \left( \varphi(\tilde{V}_j) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right) \right| = \left| \mathbb{E}_\nu \left( \varphi(\tilde{V}_j) \right) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right| \quad (6.30a)$$

$$= \left| \int_{\mathbb{R}_+} \varphi(x) d\nu_{\epsilon_0}(x) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right|. \quad (6.30b)$$

By the definition of  $\nu_{\epsilon_0}$  we can write the first integral in (6.30b) as

$$\int_{\mathbb{R}_+} \varphi(x) d\nu_{\epsilon_0}(x) = \int_{\mathbb{R}_+ \cap \Omega_{\epsilon_0}} \varphi(x) \nu(\Omega_{\epsilon_0})^{-1} d\nu(x). \quad (6.31)$$

Let us split the second integral in (6.30b) into two integrals over  $\mathbb{R}_+ \cap \Omega_{\epsilon_0}$  and  $\mathbb{R}_+ \cap \Omega_{\epsilon_0}^c$  respectively, i.e.,

$$\int_{\mathbb{R}_+} \varphi(x) d\nu(x) = \int_{\mathbb{R}_+ \cap \Omega_{\epsilon_0}} \varphi(x) d\nu(x) + \int_{\mathbb{R}_+ \cap \Omega_{\epsilon_0}^c} \varphi(x) d\nu(x). \quad (6.32)$$

## 6. ASYMPTOTIC PROPERTIES OF THE SATURATED PARTICLE FILTER

Then, by (6.31)–(6.32), (6.30b) is bounded from above by:

$$\begin{aligned} \left| \int_{\mathbb{R}_+ \cap \Omega_{\epsilon_0}} \varphi(x) \nu(\Omega_{\epsilon_0})^{-1} d\nu(x) - \int_{\mathbb{R}_+ \cap \Omega_{\epsilon_0}} \varphi(x) d\nu(x) \right| + \left| \int_{\mathbb{R}_+ \cap \Omega_{\epsilon_0}^c} \varphi(x) d\nu(x) \right| \\ \leq \|\varphi\|_\infty (|1 - \nu(\Omega_{\epsilon_0})| + \nu(\Omega_{\epsilon_0}^c)) \quad (6.33a) \\ \leq \|\varphi\|_\infty (\tilde{\epsilon} + \tilde{\epsilon}) \quad (6.33b) \\ = 2\tilde{\epsilon} \|\varphi\|_\infty \quad (6.33c) \end{aligned}$$

□

**Lemma 6.5.** *Let  $\nu$  be an arbitrary probabilistic measure on  $(\mathbb{R}_+, \Sigma_{\mathbb{R}_+})$  and let  $\tilde{\epsilon} > 0$  be a given positive constant. Furthermore, let  $\{\tilde{V}_j\}_{j=1}^N$  be a set of i.i.d. variables with a common distribution  $\nu_{\epsilon_0}$ , with  $\epsilon_0$  defined in (6.27). Finally, let  $\varphi$  be a continuous and bounded function on  $\mathbb{R}_+$ . Then, the following holds:*

$$\mathbb{E}_\nu \left( \left( \frac{1}{N} \sum_{j=1}^N \varphi(\tilde{V}_j) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right)^4 \right) \leq \frac{16}{N^2} \|\varphi\|_\infty^4 (3 + 4\tilde{\epsilon} + 6N\tilde{\epsilon}^2 + N^2\tilde{\epsilon}^4), \quad (6.34)$$

where  $\|\cdot\|_\infty$  is a supremum norm on  $\mathbb{R}_+$ .

*Proof.*

$$\mathbb{E}_\nu \left( \left( \frac{1}{N} \sum_{j=1}^N \varphi(\tilde{V}_j) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right)^4 \right) \quad (6.35a)$$

$$= \frac{1}{N^4} \mathbb{E}_\nu \left( \left( \sum_{j=1}^N \left( \varphi(\tilde{V}_j) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right) \right)^4 \right) \quad (6.35b)$$

$$= \frac{1}{N^4} \sum_{j_1, \dots, j_4=1}^N \mathbb{E}_\nu \left( \prod_{k=1}^4 \left( \varphi(\tilde{V}_{j_k}) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right) \right) \quad (6.35c)$$

Because the variables are mutually independent, the sum in (6.35c) can be decomposed into the summation of the even terms:

$$\sum_{j=1}^N \mathbb{E}_\nu \left( \varphi(\tilde{V}_j) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right)^4 \quad (6.36a)$$

$$+ 6 \times \sum_{j_1 > j_2=1}^N \mathbb{E}_\nu \left( \varphi(\tilde{V}_{j_1}) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right)^2 \times \mathbb{E}_\nu \left( \varphi(\tilde{V}_{j_2}) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right)^2, \quad (6.36b)$$

---

#### 6.4. Asymptotic Behavior of the SPF Under Improved Resampling

and the odd terms:

$$4 \times \sum_{j_1 \neq j_2=1}^N \mathbb{E}_\nu \left( \varphi \left( \tilde{V}_{j_1} \right) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right) \times \mathbb{E}_\nu \left( \varphi \left( \tilde{V}_{j_2} \right) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right)^3 \quad (6.37a)$$

$$+ 12 \times \sum_{\substack{j_1 > j_2 \geq 1 \\ j_1, j_2 \neq j_3 \geq 1}} \prod_{k=1}^2 \mathbb{E}_\nu \left( \varphi \left( \tilde{V}_{j_k} \right) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right) \times \mathbb{E}_\nu \left( \varphi \left( \tilde{V}_{j_3} \right) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right)^2 \quad (6.37b)$$

$$+ 24 \times \sum_{j_1 > \dots > j_4=1}^N \prod_{k=1}^4 \left( \mathbb{E}_\nu \left( \varphi \left( \tilde{V}_{j_k} \right) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right) \right) \quad (6.37c)$$

Since  $\nu$  is a probabilistic measure, the integral  $\left| \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right|$  is bounded from above by  $\|\varphi\|_\infty$ . Thus, for every  $j \in \{1, \dots, N\}$  and every  $k \in \mathbb{N}$  we have:

$$\mathbb{E}_\nu \left( \varphi \left( \tilde{V}_j \right) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right)^k \leq 2^k \|\varphi\|_\infty^k. \quad (6.38)$$

Furthermore, given Lemma 6.4, the first-order terms are bounded by  $2\|\varphi\|_\infty \tilde{\epsilon}$ . Therefore, by (6.36a)–(6.37c) and (6.38), the expression (6.35a) is bounded from above by:

$$\begin{aligned} \frac{2^4}{N^4} \|\varphi\|_\infty^4 \times \left( N + 6 \binom{N}{2} + 4N(N-1)\tilde{\epsilon} + 12N \binom{N-1}{2} \tilde{\epsilon}^2 + 24 \binom{N}{4} \tilde{\epsilon}^4 \right) \\ \leq \frac{2^4}{N^2} \|\varphi\|_\infty^4 (3 + 4\tilde{\epsilon} + 6N\tilde{\epsilon}^2 + N^2\tilde{\epsilon}^4) \end{aligned} \quad (6.39)$$

□

**Theorem 6.3** (Convergence of the SPF:  $\bar{c}^N$ -resampling). *Let us set  $\tilde{\epsilon} = \frac{1}{\sqrt{N}}$  and let  $\bar{c}^N$  be the resampling operator introduced in Definition 6.8. Furthermore, let  $k_k^N$  be the SPF, with the resampling operator  $\bar{c}^N$  such that the following hold:*

- I.  $K_k$  is a bounded Feller kernel,
- II.  $g_k$  is bounded, continuous and strictly positive,
- III.  $\alpha, C, z \rightarrow K_k(z|\cdot)$  and  $h_{k+1}$  are continuous functions.

Then  $k_k^N \left( \pi_{k|k}^N \right)$  converges almost surely towards the true posterior PDF  $\pi_{k+1|k+1}$ .

*Proof.* For a given  $\nu \in \mathcal{P}(\mathbb{R}_+, \Sigma_{\mathbb{R}_+})$ , by the definition of  $\bar{c}^N$  we have

$$\mathbb{E}_\nu \left( \left( \int_{\mathbb{R}_+} \varphi(x) [\bar{c}^{N,\cdot}(\nu)](dx) - \int_{\mathbb{R}_+} \varphi(x) \nu(dx) \right)^4 \right) = \mathbb{E}_\nu \left( \left( \frac{1}{N} \sum_{j=1}^N \varphi(\tilde{V}_j) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right)^4 \right), \quad (6.40)$$

where  $\{\tilde{V}_j\}_{j=1}^N$  is a set of i.i.d. random variables distributed according to  $\nu_{\epsilon_0}$ . Since  $\tilde{\epsilon} = \frac{1}{\sqrt{N}}$  then, by Lemma 6.5, we have:

$$\mathbb{E}_\nu \left( \left( \frac{1}{N} \sum_{j=1}^N \varphi(\tilde{V}_j) - \int_{\mathbb{R}_+} \varphi(x) d\nu(x) \right)^4 \right) \leq \frac{176}{N^2} \|\varphi\|_\infty^4, \quad (6.41)$$

## 6. ASYMPTOTIC PROPERTIES OF THE SATURATED PARTICLE FILTER

---

thus the resampling operator  $\bar{c}^N$  satisfies (6.7). Furthermore, by Lemmas 6.1–6.2, the kernel  $\tilde{K}_k$  is Feller. Moreover, by the definition of the operator  $\bar{c}^N$ , (6.16)–(6.17) are satisfied (with  $M_1 = \epsilon_0$  and  $M_2 = 1 - \epsilon_0$ ) hence, by Lemma 6.3, the weighted likelihood function  $\omega_k$  is bounded, continuous and strictly positive. Therefore, by Theorem 6.1,  $k_k^N \left( \pi_{k|k}^N \right)$  converges almost surely towards the true posterior PDF  $\pi_{k+1|k+1}$ .  $\square$

### 6.5 Properties of the Improved Saturated Particle Filter

Let us now discuss the practical properties of the improved Saturated Particle Filter (iSPF), i.e., the SPF with the resampling operator  $\bar{c}^N$ , in view of Theorem 6.3.

#### 6.5.1 Implementation

The conditions I)–III) of Theorem 6.3 are consistent with the conditions I)–III) of Theorem 6.2 and play exactly the same role, i.e., they ensure an “appropriate smoothness” of the model. Thanks to the construction of the operator  $\bar{c}^N$ , Condition IV) of Theorem 6.2 is not necessary anymore in Theorem 6.3. Indeed, Conditions (6.16)–(6.17) are always satisfied with  $M_1 = \epsilon_0$  and  $M_2 = 1 - \epsilon_0$ .

In practical applications at each filtering iteration we need to compute  $\epsilon_0$  according to (6.27). Since the real SPF algorithm approximates the true PDF by the set of  $N$  samples  $\{(x^i, \omega^i, q^i)\}_{i=1}^N$ , given  $\tilde{\epsilon}$ , we compute  $\epsilon_0$  by:

$$\epsilon_0 := \max \left\{ \epsilon : \sum_{i: 1-\epsilon \geq q^i \geq \epsilon} \omega^i \geq 1 - \tilde{\epsilon} \right\}. \quad (6.42)$$

A possible solution to the optimization problem (6.42) is presented in Algorithm 6.1.

---

#### Algorithm 6.1 Computation of $\epsilon_0$

---

**Input:**  $\{(\omega^i, q^i)\}_{i=1}^N, \tilde{\epsilon}$   
Initialize  $\epsilon = 0$   
**repeat**  
     $\epsilon = \epsilon + \frac{1}{N}$   
**until**  $\sum_{i: 1-\epsilon > q^i > \epsilon} \omega^i \leq 1 - \tilde{\epsilon}$   
 $\epsilon_0 = \epsilon - \frac{1}{N}$

---

With  $\epsilon_0$  chosen, the resampling procedure, represented abstractly by the operator  $\bar{c}^N$ , proceeds as follows. First, the algorithm discards the particles  $(x^i, \omega^i, q^i)$  such that either  $1 - \epsilon_0 < q^i$  or  $q^i < \epsilon_0$ . Next, the scaled degeneracy measure  $N'_{\text{eff}}$  is computed by

$$N'_{\text{eff}} := \frac{1}{\sum_{i=1}^{N'} (\omega_k^i)^2}, \quad (6.43)$$

where  $N'$  is a number of particles remaining after the discarding step. If  $N'_{\text{eff}}$  drops below a specified threshold

$$N'_T := N_T \cdot N' / N, \quad (6.44)$$

---

## 6.5. Properties of the Improved Saturated Particle Filter

---

which means that the particles that were not discarded degenerate, all the particles are resampled according to Algorithm 3.7.

If the degeneracy does not occur the algorithm resamples  $N - N'$  particles from the conditional distribution  $\nu_{\epsilon_0}$ , which is approximated by the empirical PDF

$$\left\{ \left( x^i, \frac{\omega^i}{\sum_{j=1}^{N'} \omega^j} \right) \right\}_{i=1}^{N'} . \quad (6.45)$$

Note that this resampling method has all the properties desired from the resampling algorithm discussed in Section 6.4.1. The overall resampling procedure is summarized in Algorithm 6.2.

---

**Algorithm 6.2** Saturated Particle Filter: improved resampling

---

**Input:**  $\{(x^i, \omega^i, q^i)\}_{i=1}^N, \epsilon_0, N_T$

**Discarding step:**

Discard the particles such that one of the following holds:

$$1 - \epsilon_0 < q^i \quad \text{or} \quad q^i < \epsilon_0$$

Compute the degeneracy measure for the remaining  $N'$  particles:

$$N'_{\text{eff}} := \frac{1}{\sum_{i=1}^{N'} (\omega^i)^2}$$

**Resampling step:**

**if**  $N'_{\text{eff}} < N'_T$  **then**

resample  $\{(x^i, \omega^i)\}_{i=1}^{N'}$  according to Algorithm 3.7

**else**

**for**  $i = 1$  **to**  $N - N'$  **do**

draw  $x^i$  from

$$x^i \sim \left\{ \left( x^i, \frac{\omega^i}{\sum_{i=1}^{N'} \omega^i} \right) \right\}_{i=1}^{N'}$$

$$\omega^i = \frac{1}{N - N'} \left( 1 - \sum_{i=1}^{N'} \omega^i \right)$$

**end for**

**end if**

---

As indicated in Section 6.3.2, for the proper definition of the SPF the detection function  $\alpha$  needs to be defined recursively. Indeed, from (6.24) we can see that  $\alpha$  depends on the set of probabilities of saturation  $\{q_k^i\}_{i=1}^N$ . Therefore, for the recursive selection of the appropriate detection function  $\alpha$  we proceed as follows: first, we choose an arbitrary detection function  $\alpha_0$ , and a constant  $\epsilon > 0$ ; then at each filtering step  $k = 1, 2, \dots$  we compute the minimum and maximum of the set  $\{q_k^i\}_{i=1}^N$  and set  $\alpha_k$  to be equal to

$$\alpha_k(z) := \alpha_0(z) \times \begin{cases} \min_i \{q_k^i\} (1 - \epsilon) & \text{for } z < \alpha_{k-1}^{-1}(0) \\ (1 - \max_i \{q_k^i\}) (1 - \epsilon) & \text{otherwise} \end{cases} \quad (6.46)$$

---

## 6. ASYMPTOTIC PROPERTIES OF THE SATURATED PARTICLE FILTER

---



---

**Algorithm 6.3** Update of the detection function  $\alpha_k$

---

**Input:**  $\{q_k^i\}_{i=1}^N, \alpha_0, \epsilon$

Compute minimum and maximum:

$$\text{MIN} := \min_{i=1, \dots, N} \{q_k^i\} \quad \text{MAX} := \max_{i=1, \dots, N} \{q_k^i\}$$

**if**  $z < \alpha_0^{-1}(0)$  **then**

$$\alpha_k(z) := \alpha_0(z) \text{MIN} (1 - \epsilon)$$

**else**

$$\alpha_k(z) := \alpha_0(z) (1 - \text{MAX}) (1 - \epsilon)$$

**end if**

---

This is summarized in Algorithm 6.3.

Note that the constant  $\epsilon$  used in Algorithm 6.3 can be chosen arbitrarily from the interval  $(0, 1)$ . This gives us a degree of freedom in choosing between the stronger influence of  $\alpha$  (for small values of  $\epsilon$ ) and a stricter upper bound for the weighted likelihood function  $\omega_k$  (for bigger values of  $\epsilon$ ).

The overall iSPF is summarized in Algorithm 6.4.

---

**Algorithm 6.4** improved Saturated Particle Filter

---

**Input:**  $\{(x_k^i, \omega_k^i, q_k^i)\}_{i=1}^N, \alpha_k, y_{k+1}, \epsilon, N_T$

**Output:**  $\{(x_{k+1}^i, \omega_{k+1}^i, q_{k+1}^i)\}_{i=1}^N, \alpha_{k+1}$

**Prediction:**

**for**  $i = 1$  **to**  $N$  **do**

$$\text{Compute } q_k^\alpha := q_k^i + \alpha_k (y_{k+1} - h_{k+1}(C(x_k^i)))$$

Compute  $x_{k+1}^i$  according to Algorithm 5.2

Compute  $\omega_{k+1}^i$  according to Algorithm 5.2

$$\text{Compute } q_{k+1}^i := \int_{C(x_{k+1}^i)}^{+\infty} \mathbb{P}(F(x_k, w_k) = z | x_{k+1}^i) dz$$

**end for**

**Resampling:**

Compute  $\epsilon_0$  according to Algorithm 6.1

Resample particles according to Algorithm 6.2

**for**  $i = 1$  **to**  $N$  **do**

$$\text{Compute } q_{k+1}^i := \int_{C(x_{k+1}^i)}^{+\infty} \mathbb{P}(F(x_k, w_k) = z | x_{k+1}^i) dz$$

**end for** {C} compute the predicted probabilities of saturation for the resampled particles

**Update of the detection function:**

Compute  $\alpha_{k+1}$  according to Algorithm 6.3

---

### 6.5.2 Detection Function

Let us analyze the problem of  $\alpha$  becoming negligible, mentioned in Section 6.3.2. By the definition of the resampling operator  $\bar{c}^N$  and by (6.46), for each time step  $k = 1, 2, \dots$  the image of  $\alpha_k$  contains the interval  $[-\epsilon_0(1 - \epsilon), \epsilon_0(1 - \epsilon)]$ , i.e.,  $[-\epsilon_0(1 - \epsilon), \epsilon_0(1 - \epsilon)] \subset \alpha_k(\mathbb{R})$ . Therefore,  $\alpha_k$  is never trivial. However, the value of  $\epsilon_0$  depends on the value of  $\tilde{\epsilon}$  (see (6.27)),

## 6.5. Properties of the Improved Saturated Particle Filter

which in view of Theorem 6.3, decreases with the rate  $\frac{1}{\sqrt{N}}$  when  $N \rightarrow \infty$ . This means that the measure  $\nu_{\epsilon_0}$  of the set  $\Omega_{\epsilon_0}$  increases, hence by (6.25)–(6.27)  $\epsilon_0$  decreases, and therefore the image of  $\alpha_k$  becomes narrower.

The rate of decrease of  $\epsilon_0$  depends on the particular shapes of both the distribution of the probabilities of saturation  $\{q_k^i\}_{i=1}^N$  and of the distribution of weights  $\{\omega_k^i\}_{i=1}^N$ . The following example illustrates this dependency. First, let us assume that both sets  $\{q_k^i\}_{i=1}^N$  and  $\{\omega_k^i\}_{i=1}^N$  are uniformly distributed on the interval  $[0, 1]$ . Then, for a given number of particles  $N$ , we expect (in the statistical sense) to discard  $\sqrt{N}$  particles and the expected value of  $\epsilon_0$  is equal to  $\frac{1}{2\sqrt{N}}$ . Second, let us assume that the weights  $\{\omega_k^i\}_{i=1}^N$  are again distributed uniformly on the interval  $[0, 1]$ , but the set  $\{q_k^i\}_{i=1}^N$  is approximated by the Gaussian distribution<sup>1</sup> with mean  $\frac{1}{2}$  and the standard deviation  $\frac{1}{10}$ . In such a case, we still expect to discard  $\sqrt{N}$  particles at each step, but this time the value of  $\epsilon_0$  is given by  $\frac{1}{2} + \frac{\sqrt{2}}{10} \operatorname{erf}^{-1}\left(\frac{2}{\sqrt{N}} - 1\right)$ , where  $\operatorname{erf}$  is the error function defined by:

$$\operatorname{erf}(z) := \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt.$$

Table 6.1 compares the two cases for three different values of  $N$ .

Table 6.1: Results obtained by the SPF for different distributions of probabilities of saturation  $\{q_k^i\}_{i=1}^N$  and weights  $\{\omega_k^i\}_{i=1}^N$

$\{q_k^i\}_{i=1}^N, \{\omega_k^i\}_{i=1}^N \approx \mathcal{U}(0, 1)$			
	$N = 10^2$	$N = 10^4$	$N = 10^6$
expected number of discarded particles	10	100	1000
expected value of $\epsilon_0$	0.05	0.005	0.0005
$\{q_k^i\}_{i=1}^N \approx \mathcal{N}\left(\frac{1}{2}, \frac{1}{10}\right), \{\omega_k^i\}_{i=1}^N \approx \mathcal{U}(0, 1)$			
	$N = 10^2$	$N = 10^4$	$N = 10^6$
expected number of discarded particles	10	100	1000
expected value of $\epsilon_0$	0.372	0.267	0.191

As we can see the expected value of  $\epsilon_0$  is strongly dependent on the distribution of  $\{q_k^i\}_{i=1}^N$ . Therefore, the influence of  $\alpha_k$ , which is determined by the value of  $\epsilon_0$ , also depends on the shape of  $\{q_k^i\}_{i=1}^N$ . Let us explain the nature of this dependency by analyzing the results from Table 6.1.

In the first case, where  $\{q_k^i\}_{i=1}^N \approx \mathcal{U}(0, 1)$  (where  $\approx$  denotes being distributed according to a given distribution), the model assigns the same probability to all the possible values of the probabilities of saturation  $q_k^i$ . In such situations the standard Bayesian update procedure should be more than sufficient in obtaining an accurate estimate. Thus, the small values of  $\epsilon_0$ , and therefore the low influence of  $\alpha_k$  is acceptable.

In the second case, where  $\{q_k^i\}_{i=1}^N \approx \mathcal{N}\left(\frac{1}{2}, \frac{1}{10}\right)$ , most of  $q_k^i$  are close to  $\frac{1}{2}$ . This means that the model is very uncertain in predicting whether the saturation will occur or not. In such cases the standard Bayesian approach is slow in detecting extreme changes of the system

<sup>1</sup>The set  $\{q_k^i\}_{i=1}^N$  is bounded, therefore by saying that it is approximated by Gaussian distribution we mean a Gaussian with truncated tails and appropriately rescaled.

## 6. ASYMPTOTIC PROPERTIES OF THE SATURATED PARTICLE FILTER

(e.g., saturation). Thus, for this example, it is strongly recommended to enforce the update procedure. For that we require a strong influence of  $\alpha_k$ , hence relatively large values of  $\epsilon_0$ .

Note that in both cases the expected number of discarded particles is small compared to the total number of particles. Therefore, the problem of losing diversity of the samples [Arulampalam et al., 2002] is avoided.

### 6.5.3 Numerical Example

We finish this section with a comparison of the SPF developed in Chapter 5, the iSPF derived in this chapter, and the BPF discussed in Chapter 3. To illustrate the abilities of all three methods we use the SSDS similar to the one discussed in Section 5.3.3, given by

$$x_{k+1} = \min(x_k + w_k, C(x_k)), \quad (6.47a)$$

$$y_k = x_k + v_k, \quad (6.47b)$$

where  $w_k$  is a random variable distributed according to the exponential distribution, with parameter  $\theta$ , i.e., with the expected value  $\mathbb{E}w_k = \theta^{-1}$ . The variable  $v_k$  is a zero-mean Gaussian variable with standard deviation  $\sigma_v$ . The boundary function  $C(\cdot)$  is defined by:

$$C(x) := x + \log(2)/\theta. \quad (6.48)$$

The state model (6.47a) is nonlinear and non-Gaussian, whereas the observation model (6.47b) is both linear and conditionally Gaussian.

To simulate the process from the initial state  $x_0 = 1$ , we used  $\theta = 1$ ,  $\sigma_v = 1$ . The length of the simulation is 20 time steps. Note that because the variable  $w_k$  is exponentially distributed, the cumulative density function of the random variable  $x_{k+1}$  is known. Thus, the integral in (6.10) can be computed analytically:

$$q_k^i = \exp(-\theta(C(x_k^i) - x_k^i)). \quad (6.49)$$

Figure 6.1 compares the results obtained by applying the BPF, the iSPF, and the SPF with 10 and 1000 particles respectively. All three filters use the model (6.47a)–(6.47b) with true parameters. The offset of 0.5 is introduced by setting the initial state  $p_0$  for all three filters to  $p_0(\cdot) = \mathcal{N}(\cdot; 0.5, 0.1)$ . The resampling threshold  $N_T$  is set to 30% of the number of particles. Both the iSPF and the SPF are using the same detection function  $\alpha_0$  given by:

$$\alpha(z) = \begin{cases} 1 & \text{if } z > 2, \\ -1 & \text{if } z < 0, \\ z - 1 & \text{otherwise,} \end{cases} \quad (6.50)$$

where  $z = y_{k+1} - h_{k+1}(C(x_k^i))$  is evaluated at each time step.

Figure 6.1 presents the average of ten independent filters of each type applied to the simulated signal. From both figures we can conclude that the new resampling procedure improves the performance of the SPF introduced in Chapter 5. However, with the growing number of samples the difference between the SPF and the iSPF becomes smaller. This is not surprising since in the view of Theorems 6.2 and 6.3 both filters converge to the same distribution as the number of samples increase.

Both simulations present results for a relatively small number of particles. This is because, as it was previously explained the influence of the detection function is the most visible when there are few particles. In this example we can observe that both the iSPF and the SPF

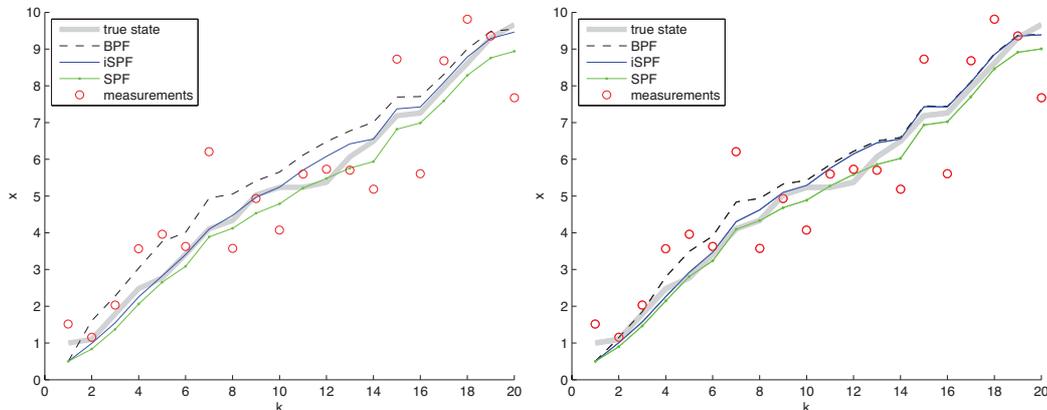


Figure 6.1: The BPF, the iSPF and the SPF with 10 particles (left) and 1000 particles (right) applied to the system (6.47a)–(6.47b). The thick solid line is the true value of the state, the circles denote the measurements, the thin dashed line denotes the MMSE estimate obtained by the BPF, the thin solid line represents the MMSE estimate of the state obtained by the iSPF and the thin solid dotted line represents the MMSE estimate of the state obtained by the SPF.

outperform the BPF. This is confirmed by comparing the *mean square errors*<sup>1</sup> (MSE) of the three methods. Table 6.2 reports values of such errors for 10, 100, and 1000 particles.

Table 6.2: The MSE of three filters: the BPF, the SPF and the iSPF obtained for different numbers of particles  $N = 10, 100, 1000$ .

	$N = 10$	$N = 100$	$N = 1000$
BPF	0.3726	0.1808	0.1848
SPF	0.1516	0.0990	0.0946
iSPF	0.0693	0.0933	0.0880

Finally, Figure 6.2 shows the standard deviations of the MMSE estimates of three discussed filters. It might be noticed that the spread of the MMSE estimates of filters using 10 particles is much higher than the spread of the MMSE estimates of filters using 1000 particles. In the latter case the spread is negligible, and in the former case it is considerable only for the BPF during the first few steps of the simulation. Thus, the result of the simulation suggests that the BPF, the SPF and the iSPF applied to the system (6.47) achieve similar performance in terms of precision of the estimates.

## 6.6 Conclusions

In this chapter we have derived the *improved Saturated Particle Filter* (iSPF) which, similarly to the Saturated Particle Filter (SPF), exploits a specific structure of the Saturated Stochastic Dynamical Systems (SSDS) to improve the accuracy of the estimates. We have discussed

<sup>1</sup>By the MSE we understand the average squared deviation of the estimate from the true value of the state, i.e.,  $\sum_{k=1}^{20} (\hat{x}(k) - x(k))^2 / 20$ .

## 6. ASYMPTOTIC PROPERTIES OF THE SATURATED PARTICLE FILTER

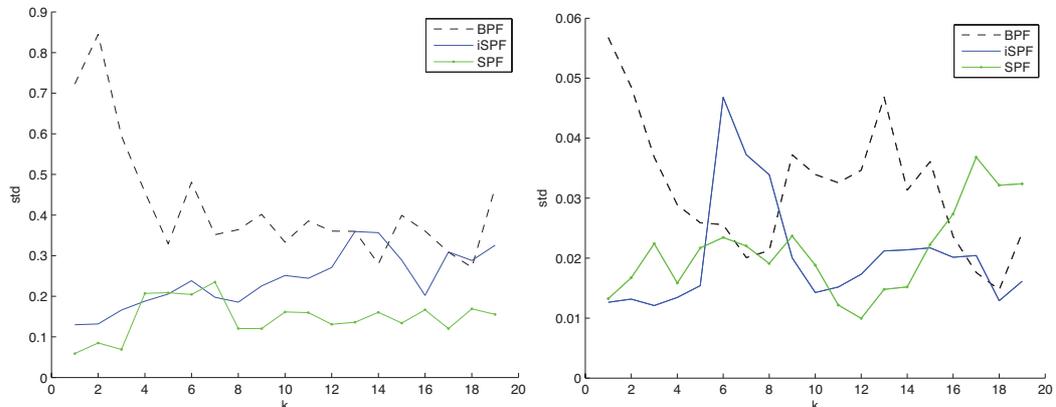


Figure 6.2: The standard deviations of 10 MMSE estimates obtained by: the BPF (dashed line), the iSPF (solid line) and the SPF (solid-dotted line) with 10 particles (left) and 1000 particles (right) applied to the system (6.47a)–(6.47b). Note that the standard deviations of the filter employing 1000 particles is one order of magnitude lower than the standard deviations of filters using 10 particles.

asymptotic properties of the SPF and the iSPF separately and we have given the conditions under which each method converges to the optimal filter.

### Summary

A characteristic feature of both filters is the incorporation of the measurement in the prediction step of the Bayesian filtering. This is done through the use of the detection function  $\alpha$ . The advantage of the iSPF over the SPF comes from the novel resampling algorithm that makes the former method more accurate and computationally more efficient. Furthermore, the introduction of the aforementioned resampling makes the iSPF applicable for essentially any type of one-dimensional SSDS, whereas the applicability of the SPF is limited to the class of systems satisfying conditions of Theorem 6.2.

In Section 6.3 we have formulated the SPF as an operator acting on the spaces of probability measures  $\mathcal{P}(\mathbb{R}_+)$ . Such an abstraction enables the proof of the almost sure convergence of the posterior PDF given by the standard SPF to the true posterior PDF. We have also discussed the practical advantages and shortcomings of the standard SPF.

In Section 6.4 we have derived the iSPF. This new algorithm is different from the standard SPF in two aspects:

- I. A novel resampling procedure  $\bar{c}^N$  is used to discard the particles with low weights such that the corresponding probability of saturation  $q^i$  achieves extreme values ( $q^i \approx 0$  or  $q^i \approx 1$ ),
- II. The detection function  $\alpha_k$  is updated recursively at each time step  $k$ .

The resampling method  $\bar{c}^N$  helps reducing the computational load of the filter by preventing expensive numerical operations on the particles with negligible weights. The dynamic nature of the detection function increases the flexibility of the filter making it applicable to a wider class of dynamical systems. We have shown in Theorem 6.3 that the iSPF also converges almost surely to the true posterior PDF.

### Discussion

From Theorem 6.3 we have concluded that as the number of samples  $N$  grows the influence of the detection function  $\alpha_k$  declines. This is not a surprise since the BPF can be considered to be a SPF with a detection function  $\alpha^{BPF} = 0$ . Furthermore, both the iSPF and the BPF converge almost surely to the same distribution. Thus, with the growing number of samples the difference between these two estimators becomes smaller, hence the distance between  $\alpha_k$  and  $\alpha^{BPF}$  also converges to zero.

The influence of the detection function  $\alpha_k$  is most noticeable when the number of samples  $N$  is relatively small. The strength of the influence always depends on the distribution of the weights  $\{\omega_k^i\}_{i=1}^N$  and the distribution of the probabilities of saturation  $\{q_k^i\}_{i=1}^N$ . An analysis of this dependency has been illustrated on an example described in Table 6.1. We have concluded that the influence of  $\alpha_k$  is much stronger in the case when the set  $\{q_k^i\}_{i=1}^N$  is concentrated around  $\frac{1}{2}$ , i.e.,  $\{q_k^i\}_{i=1}^N \approx \mathcal{N}(\frac{1}{2}, \frac{1}{10})$ , than in the case when the distribution of the probabilities of saturation  $\{q_k^i\}_{i=1}^N$  is heavy tailed, i.e.,  $\{q_k^i\}_{i=1}^N \approx \mathcal{U}(0, 1)$ . Furthermore, in Section 7.3.2 the illustrative example of SSDS is used to compare the performance of the iSPF, the SPF and the BPF.

### Further Research

In general, the influence of the detection function  $\alpha_k$  can be further modified by an appropriate choice of the constant  $\epsilon > 0$  in Algorithm 6.3. This must be done carefully because, by (6.21), the value of  $\epsilon$  determines the upper bound for the weighted likelihood function  $\omega_k$ , hence also the variance of the weights  $\{\omega_k^i\}_{i=1}^N$ . The exact nature of these relations is a matter of the ongoing research.

## 6. ASYMPTOTIC PROPERTIES OF THE SATURATED PARTICLE FILTER

## Chapter 7

# Solution to the Hopper Estimation Problem

Parts of this chapter were published in:

- “*Particle Filters for the Estimation of the Average Grain Diameter of the Material Excavated by a Hopper Dredger*”, Proceedings of the IEEE Conference on Control Applications, Yokohama 2010, pp. 292-297, Paweł Stano, Zsófia Lendek, Robert Babuška, Jelmer Braaksma, and Cees de Keizer.
- “*Estimation of the Soil-Dependent Time-Varying Parameters of the Hopper Sedimentation Model: the FPF versus the BPF*”, in press, Paweł Mirosław Stano, Adam K. Tilton, and Robert Babuška.

### Abstract

The Hopper sedimentation process describes the settling of the material excavated by the Drag-Head and transported through the pipeline into the hopper. The settling rate of the material strongly depends on the type of soil that was pumped into the hopper. The most important soil-dependent parameter of the sedimentation process is the average grain diameter  $d_m$  of the excavated soil. The accurate knowledge of  $d_m$  and the codependent variables such as the sand bed height  $h_s$ , sand bed mass  $m_s$ , and the mixture density  $\rho_m$  is necessary to control the sedimentation process in an optimal way. These variables need to be estimated online to be integrated into the automated controller. In this chapter we discuss the algorithms that compute the estimates of the aforementioned variables. To find an optimal solution we split the estimation problem into several separate scenarios. We analyze them independently and for each we recommend filtering algorithm.

### 7.1 Introduction

In Chapter 2 the dynamical sedimentation model was presented and the estimation problem was formulated. To summarize, the continuous-time dynamical system that models the sedi-

## 7. SOLUTION TO THE HOPPER ESTIMATION PROBLEM

---

mentation process is given by

$$\dot{m}_s = f_e(d_m, h_t, h_s, Q_o) f_s(d_m, m_t, h_s, h_t) \rho_s(d_m), \quad (7.1a)$$

$$\dot{h}_s = \frac{f_e(d_m, h_t, h_s, Q_o) f_s(d_m, m_t, h_s, h_t)}{A}, \quad (7.1b)$$

$$\dot{h}_t = \frac{Q_i - Q_o}{A}, \quad (7.1c)$$

$$\dot{m}_t = Q_i \rho_i - Q_o \rho_o, \quad (7.1d)$$

and the estimation objective is to obtain online estimates of the sand bed height  $h_s$ , sand bed mass  $m_s$ , average grain diameter  $d_m$ , and the mixture density  $\rho_m$  from the available measurements.

In this chapter we show how these problems can be solved. To find an optimal solution we split the problem stated above into several distinct scenarios and analyze them independently. Namely, we divide the sedimentation process into three operational modes, which arise naturally during the dredging. These are:

- I. Mode 1: The No-Overflow loading phase,
- II. Mode 2: The Overflow loading phases with weak erosion influence,
- III. Mode 3: The Overflow loading phases with strong erosion influence.

By considering each of the above modes separately we are able to analyze several estimation techniques which, properly assigned to a suitable scenario, increase the efficiency of the estimation in terms of accuracy and speed. Namely, we design four estimators:

- I. Reduced-Order Particle Filter (ROPF),
- II. Bootstrap Particle Filter (BPF),
- III. Feedback Particle Filter (FPF),
- IV. improved Saturated Particle Filter (iSPF).

Each of the aforementioned methods is characterized by different stochastic and numerical properties. In Section 7.2 we derive the ROPF which proves to be an accurate solution of the Hopper Estimation Problem for Mode 2. The ROPF is later used as a benchmark method in Section 7.4 where the iSPF is applied to the Hopper Estimation Problem for Mode 3. Section 7.3 presents the applications of the FPF to the Hopper Estimation Problem for Mode 1. The performance of the FPF is compared to the performance of the benchmark solutions provided by the BPF. Section 7.5 concludes the chapter by providing the recommended solution to the Hopper Estimation Problem for all the operational modes and the suggested directions for the future research.

### 7.2 Overflow Loading Phases with Weak Erosion: the Reduced-Order Particle Filter

In this section we derive a discrete-time simplified stochastic dynamical model of the sedimentation process. Such a reduced-order system serves as a basis for the ROPF, which is a very fast algorithm used to estimate the sand bed height  $h_s$ , the sand bed mass  $m_s$  and the average grain diameter  $d_m$ . In order to reduce the complexity of the complete sedimentation model (7.1), which is discussed in detail in Chapter 2, we make the following assumptions:

## 7.2. Overflow Loading Phases with Weak Erosion: the Reduced-Order Particle Filter

---

- I. We consider the total hopper mass  $m_t$  to be a known input to the system rather than a state of the system.
- II. In the dynamical model of the sand bed mass  $m_s$  we approximate the nonlinear function  $Q_s$  by a linear function of the sand bed height  $h_s$ .
- III. We consider the total height  $h_t$  to be a stochastic input to the system with the associated uncertainty included into the state equations.

The first assumption is justified by the fact that the total hopper mass  $m_t$  is derived from the draught of the ship, which is a function of the ship mass [Braaksma, 2008]. The draught is calculated from the pressure sensors located in the bottom of the hull, which are very accurate. The mass of the hopper is computed by subtracting the mass of an empty ship from the mass of a full ship. The second assumption is justified by the dynamical relation between the  $m_s$  and  $h_s$  (2.27) and the slowly varying dynamics of the sand bed height  $h_s$ . The third assumption is justified by the fact that the total height  $h_t$  is measured on board of the ship.

Two variables are measured with uncertainty: the total height of the mixture in the hopper  $h_t$  and the height of the sand bed  $h_s$ . Both are assumed to be corrupted by zero-mean, time-invariant Gaussian noises  $e_t^o$  and  $e_s^o$  with standard deviations  $\sigma_t^o$  and  $\sigma_s^o$  respectively. Two more variables are assumed to be known inputs to the system: the outgoing flow rate  $Q_o$  and the total mass  $m_t$ , which are assumed to be measured with negligible noises [Braaksma, 2008; Lendek et al., 2008]. The measurements are taken with a sampling period  $T_s$ .

We show that to solve the Hopper Estimation Problem, the ROPF requires measurements of the mixture density  $\rho_m$ . This is possible only during the overflow loading phases. Indeed, experimental data suggest [Braaksma, 2008] that the mixture density  $\rho_m$  is equal to the overflow density  $\rho_o$ . We show how this extra information is used by the ROPF. Furthermore, we show that the ROPF that does not have access to the measurements of  $\rho_m$  fails to produce accurate estimates. This means that the ROPF cannot be used during the No-Overflow loading phase since the measurements of the mixture density  $\rho_m$  cannot be obtained.

### 7.2.1 Derivation of the Discrete-Time Stochastic System

In this section we derive the reduced-order discrete-time model that is used by the ROPF to estimate the sand bed height  $h_s$ , the sand bed mass  $m_s$ , and the average grain diameter  $d_m$ .

The derivative of  $h_s$  at time step  $k$  is approximated by applying the first order Euler discretization:

$$\frac{d}{dt}h_{s,k} \approx \frac{h_{s,k} - h_{s,k-1}}{T_s} + e_s, \quad (7.2)$$

where the last term is an approximation error (time invariant, zero-mean Gaussian with standard deviation  $\sigma_s$ ) and  $T_s$  is the sampling period. Combining (7.1b) with (7.2) yields the dynamics of the sand bed height  $h_s$ :

$$h_{s,k+1} = h_{s,k} + T_s e_s + \frac{T_s}{A} f_e(d_{m,k}, h_{t,k}, h_{s,k}, Q_{o,k}) f_s(d_{m,k}, \rho_{m,k}), \quad (7.3)$$

where  $f_e$  and  $f_s$  are functions describing the erosion and the settling processes respectively. The erosion factor  $f_e$ , introduced in (2.23), in discrete-time variables is given by:

$$f_e(d_{m,k}, h_{t,k}, h_{s,k}, Q_{o,k}) = \max\left(1 - \frac{Q_{o,k}^2}{(k_e(d_{m,k})(h_{t,k} - h_{s,k}))^2}, 0\right). \quad (7.4)$$

## 7. SOLUTION TO THE HOPPER ESTIMATION PROBLEM

---

The settling function  $f_s$ , derived from (2.24) by considering the mixture density  $\rho_m$  as a separate variable, takes a simplified form:

$$f_s(d_{m,k}, \rho_{m,k}) = Av_{s0}(d_{m,k}) \frac{\rho_{m,k} - \rho_w}{\rho_s(d_{m,k}) - \rho_{m,k}} \left( \frac{\rho_q - \rho_{m,k}}{\rho_q - \rho_w} \right)^{\beta(d_{m,k})}, \quad (7.5)$$

where the functions  $k_e, \rho_s, v_{s0}$  and  $\beta$  model the soil dependent parameters as functions of the average grain diameter  $d_m$  as described in Section 2.4.

The variable  $\rho_{m,k}$  is in fact a function of the variables  $m_{t,k}, m_{s,k}, h_{s,k}$ , and  $h_{t,k}$ :

$$\rho_{m,k}(m_{t,k}, h_{s,k}, h_{t,k}, d_{m,k}) = \frac{m_{t,k} - Ah_{s,k}\rho_s(d_{m,k})}{Ah_{t,k} - Ah_{s,k}}. \quad (7.6)$$

It is possible to further reduce the nonlinearity of the ROPF, e.g., by proper preprocessing of the mixture density  $\rho_{m,k}$  before each filtering step. The relation between the mixture density  $\rho_{m,k}$  and the performance of the filtering algorithm has been a subject of extensive studies. The optimal strategy of how to handle  $\rho_{m,k}$  depends on multiple factors such as the excavated soil type, the loading phase or the types of sensors available on board, to name a few. In Section 7.2.2 we investigate two scenarios:

- I. Scenario A: the mixture density  $\rho_m$  is known,
- II. Scenario B: the mixture density  $\rho_m$  is unknown.

The dynamics of  $m_s$  are obtained by noticing that by (7.1a)–(7.1b) we have:

$$\dot{m}_s = A\rho_s(d_m)\dot{h}_s. \quad (7.7)$$

Euler discretization of (7.7) combined with (7.2) leads to:

$$m_{s,k+1} = m_{s,k} + A\rho_s(d_{m,k})(h_{s,k} - h_{s,k-1}) + T_s A\rho_s(d_{m,k})e_s + e_m, \quad (7.8)$$

where  $e_m$  is a zero mean Gaussian noise with standard deviation  $\sigma_m$ .

Note, that in (7.8) the derivative of the discretized sand bed height  $\frac{d}{dt}h_{s,k}$  is shifted one time step backwards. This approach is justified by the slowly varying nature of the signal  $h_s$ .

The evolution of the average grain diameter  $d_{m,k}$  is described by a discrete-time random-walk model:

$$d_{m,k+1} = d_{m,k} + e_d, \quad (7.9)$$

where  $e_d$  is a zero-mean Gaussian random variable with standard deviation  $\sigma_d$ . This is an approximation of the continuous-time model presented in Section 2.5.

The variable  $h_t$  is measured with the measurement error  $e_t^o$ . Because we are not concerned with the estimation of  $h_t$  and we aim to reduce the order of the system we consider  $h_t$  as an input variable and include the associated uncertainty  $e_t^o$  into the state equation (7.3) as a noise. Then the state, input, and output vectors of the system are given by:

$$\mathbf{x} = \begin{pmatrix} m_s \\ h_s \\ d_m \end{pmatrix}, \quad \mathbf{u} = \begin{pmatrix} m_t \\ Q_o \\ h_t \end{pmatrix}, \quad y = h_s.$$

The final form of the state-space model in these variables is expressed as:

$$x_{1,k+1} = x_{1,k} + A\rho_s(x_{3,k})(x_{2,k} - x_{2,k-1}) + T_s A\rho_s(x_{3,k})e_s + e_m, \quad (7.10a)$$

$$x_{2,k+1} = x_{2,k} + T_s e_s + T_s G(\mathbf{x}_k, \mathbf{u}_k, e_t^o), \quad (7.10b)$$

$$x_{3,k+1} = x_{3,k} + e_d, \quad (7.10c)$$

$$y_k = x_{2,k} + e_s^o, \quad (7.10d)$$

## 7.2. Overflow Loading Phases with Weak Erosion: the Reduced-Order Particle Filter

---

where the function  $G$  is given by:

$$G(\mathbf{x}_k, \mathbf{u}_k, e_t^o) = \frac{f_e(x_{3,k}, u_{3,k} + e_t^o, x_{2,k}, u_{2,k}) f_s(x_{3,k}, \rho_{m,k}(\mathbf{x}_k, \mathbf{u}_k))}{A}, \quad (7.11)$$

and the variable  $\rho_{m,k}$  is given by:

$$\rho_{m,k}(\mathbf{x}_k, \mathbf{u}_k) = \frac{u_{1,k} - Ax_{2,k}\rho_s(x_{3,k})}{Au_{3,k} - Ax_{2,k}}. \quad (7.12)$$

### Stochastic Properties of the System

It is assumed that errors  $e_s$  and  $e_m$  are independent and zero-mean Gaussians with variances  $\sigma_s^2$  and  $\sigma_m^2$  respectively. Consequently, the random variable  $T_s A \rho_s(x_{3,k}) e_s + e_m$  is also a zero-mean Gaussian with the variance  $\sigma_{x_1}^2$  given by

$$\sigma_{x_1}^2 = (T_s A \rho_s(x_{3,k}) \sigma_s)^2 + \sigma_m^2. \quad (7.13)$$

The random variable  $e_t^o$  influences only the erosion part  $f_e$  of the variable  $G$ . Therefore, since during the first phase  $f_e$  is constant,  $G(\mathbf{x}_k, \mathbf{u}_k)$  in (7.10b) becomes a deterministic function of state and observation. Thus, by (7.5), as long as there is no overflow ( $Q_o = 0$ ), the variable  $T_s e_s + T_s G(\mathbf{x}_k, \mathbf{u}_k)$  is normally distributed with mean  $\mu_{x_2}$  and standard deviation  $\sigma_{x_2}$  given by:

$$\mu_{x_2} = T_s v_{s0}(x_{3,k}) \frac{\rho_{m,k} - \rho_w}{\rho_s(x_{3,k}) - \rho_{m,k}} \left( \frac{\rho_q - \rho_{m,k}}{\rho_q - \rho_w} \right)^{\beta(x_{3,k})}, \quad (7.14a)$$

$$\sigma_{x_2} = T_s \sigma_s. \quad (7.14b)$$

The probabilistic model (7.10b) for the constant-volume phase is more involved. The error  $T_s e_s$  and the random variable  $T_s G(\mathbf{x}_k, \mathbf{u}_k, e_t^o)$  are independent, but the latter is not Gaussian. Thus, the probability density function (PDF)  $p_{x_2}$  of the sum of these variables is a convolution of their PDFs ( $p_{x_2,e}$  and  $p_{x_2,G}$  respectively):

$$p_{x_2}(y) = \int p_{x_2,e}(y-z) p_{x_2,G}(z) dz. \quad (7.15)$$

The PDF of the normally distributed variable  $T_s e_s$  is known. In order to derive the PDF of  $T_s G(\mathbf{x}_k, \mathbf{u}_k, e_t^o)$  we use the following proposition:

**Proposition 7.1.** *If  $X$  is a normally distributed random variable with mean  $\mu$  and variance  $\sigma^2$  and  $C_1, C_2$  are certain positive constants then the probability density function of the variable*

$$C_1 \max\left(0, 1 - \frac{C_2}{X^2}\right) \quad (7.16)$$

is given by

$$p_{C_1 \max(0, 1 - \frac{C_2}{X^2})}(x) = \sqrt{\frac{C_1 C_2}{2\pi\sigma^2(C_1 - x)^3}} e^{-\frac{C_1 C_2}{C_1 - x} + \frac{\mu^2}{2\sigma^2}} \cosh\left(\frac{\mu\sqrt{C_1 C_2}}{\sigma^2\sqrt{C_1 - x}}\right) \mathbf{1}_{(0, C_1]}(x) \quad (7.17a)$$

$$+ \left(1 - \int_0^1 \sqrt{\frac{C_2}{2\pi\sigma^2 y^3}} e^{-\frac{y}{2\sigma^2} + \frac{\mu^2}{2\sigma^2}} \cosh\left(\frac{\mu\sqrt{C_2}}{\sigma^2\sqrt{y}}\right) dy\right) \delta_0(x), \quad (7.17b)$$

where  $\mathbf{1}_{(0, C_1]}$  is an indicator function and  $\delta_0$  is the Dirac delta.

## 7. SOLUTION TO THE HOPPER ESTIMATION PROBLEM

The proof of Proposition 7.1 can be found in Appendix B. From Proposition 7.1 we can define the PDF of the stochastic model (7.10b). At each time step  $k$  we define

$$X_k = k_e(d_{m,k})(e_t^o + h_{t,k} - h_{s,k}), \quad (7.18a)$$

$$C_{1,k} = T_s v_{s0}(d_{m,k}) \frac{\hat{\rho}_{o,k} - \rho_w}{\rho_s(d_{m,k}) - \hat{\rho}_{o,k}} \left( \frac{\rho_q - \hat{\rho}_{o,k}}{\rho_q - \rho_w} \right)^{\beta(d_{m,k})}, \quad (7.18b)$$

$$C_{2,k} = Q_{o,k}^2. \quad (7.18c)$$

Such an  $X_k$  is a normally distributed random variable with mean  $\mu_k$  and variance  $\sigma_k^2$  given by:

$$\mu_k = k_e(d_{m,k})(h_{t,k} - h_{s,k}), \quad (7.19a)$$

$$\sigma_k^2 = k_e(d_{m,k})^2 \sigma_t^2, \quad (7.19b)$$

where  $\sigma_t^2$  is the variance of the observation noise  $e_t^o$ . Furthermore, both  $C_{1,k}$  and  $C_{2,k}$  defined above are positive, and, therefore, the PDF of  $T_s G(\mathbf{x}_k, \mathbf{u}_k, e_t^o)$  follows by applying Proposition 7.1.

From the presented description it can be seen that the system (7.10) exhibits severe nonlinearities (7.3)–(7.6) together with non-Gaussian probabilistic behavior (7.17). Due to this, the parametric filters, such as those described in Chapter 3.3, are not suitable for this system. Therefore, to estimate the desired average grain diameter  $d_m$  we use nonparametric methods. The results of applying the PF are discussed in Section 7.2.2.

### 7.2.2 Numerical Simulations

#### Setting

The bootstrap PF described in Chapter 3 using the reduced-order model (7.10) is applied to simulated data, generated using the probabilistic sedimentation model (7.1). The sampling time  $T_s$  is set to 1[s]. We simulated each of the loading phases independently. In each case, the value of the average grain diameter  $d_m$  is changed twice during the phase. In the real process, this corresponds to approaching a dredging area with a different type of in situ material. To analyze the tracking properties of the filter, each change in  $d_m$  is set to be step-like, which models dramatic changes in the soil-type of the dredged material (during the real dredging operation these transitions are generally smoother). The loading of the hopper is illustrated in Figure 7.1a (No-Overflow and Constant Volume phases) where also the time evolution of the total height of the mixture  $h_t$  and the sand bed height  $h_s$  are presented.

The hopper used for the simulations is of a rectangular parallelepiped form with the base area  $A = 600[\text{m}^2]$ . Furthermore, it is assumed that the excavated soil is sand with the mean grain diameter  $d_m$  varying between 0.35[mm] and 0.7[mm]. We use the ROPF with  $N = 1000$  particles and the resampling threshold  $N_T$  that was experimentally set to 500 (i.e., 50% of the number of particles  $N$ ). These, and the standard deviations of the noises used by model (7.10) are summarized in Table 7.1

Table 7.1: Parameters of the ROPF

Number of particles	Resampling threshold	Initial offset in $\hat{d}_m$ [mm]	Standard deviations in process noises			Standard deviations in observation noises	
			$\sigma_m$ [tons]	$\sigma_d$ [mm]	$\sigma_s$ [m]	$\sigma_t^o$ [m]	$\sigma_s^o$ [m]
1000	500	+0.3	1	0.1	0.001	0.1	0.05

## 7.2. Overflow Loading Phases with Weak Erosion: the Reduced-Order Particle Filter

---

In the prediction step of the PF designed for the constant-volume phase, it is required, among others, to sample from the random variable defined in (7.10b). However, drawing from the PDF given by Proposition 7.1 is in general not straightforward. Therefore, for each particle  $x_{2,k-1}^i$  an approximation of the true random sample is generated by Algorithm 7.1.

---

### Algorithm 7.1 Approximate sampling

---

**Input:**  $x_{2,k-1}, \mu_{k-1}, \sigma_{k-1}, C_{1,k-1}, C_{2,k-1}, T_s, \sigma_s$   
**for**  $i = 1, 2, \dots, N$  **do**  
    Draw two independent samples:  
     $x_1^i \sim \mathcal{N}(x_{2,k-1}^i, T_s \sigma_s)$   
     $x_2^i \sim \mathcal{N}(\mu_{k-1}^i, \sigma_{k-1}^i)$   
    Perform nonlinear transformation:  
     $\tilde{x}_2^i = T_s C_{1,k-1}^i \max(0, 1 - C_{2,k-1}^i / x_2^i)$   
    Assign approximated sample:  
     $x_{2,k}^i = x_1^i + \tilde{x}_2^i$   
**end for**

---

### Results for Scenario A ( $\rho_m$ is known)

We start by investigating the situation where the mixture density  $\rho_m$  is a known input to the system. We simulate the sedimentation process for two loading phases: the No-Overflow phase and the Constant-Volume phase. We do not investigate the filters performance during the Constant-Tonnage loading phase because during that phase the dynamical and stochastic properties of the system (7.10) are qualitatively the same as for the Constant-Volume phase.

We present the results obtained by the average of ten filters running in parallel. A comparison of the estimated and simulated signals of the average grain diameter  $d_m$  for the No-Overflow phase and Constant-Volume phase is presented in Figure 7.1b.

From Figure 7.1b it can be observed that the ROPF works very well and that its performance in the two loading phases is comparable. In each case, the convergence times are approximately 40[s], 20[s], and 20[s], respectively, which is acceptable for the TSHD's controller developed in [Braaksma, 2008]. Moreover, as has been stated before, in real dredging operations the changes in the average grain diameter are smoother, thus the convergence of the filter is expected to be further improved.

For both phases the estimate is slightly biased. However, the absolute estimation errors are smaller than the standard deviation  $\sigma_d$  of the noise that corrupts the estimated state  $d_m$ . This makes the errors acceptable. In fact for higher values of  $d_m$  the bias is negligible. This is confirmed by analyzing the *residuals*, i.e., the difference between the estimate and the true state, of the estimate  $\hat{d}_m$  in steady state:

$$\text{Res}(\hat{d}_m) := (\hat{d}_m - d_m). \quad (7.20)$$

Because the simulated  $d_m$  is piecewise constant with three distinct values corresponding to different soil type, it is reasonable to compare the residuals  $\text{Res}(\hat{d}_m)$  separately for each  $d_m = 0.35, 0.7, 0.45$ [mm]. To have a fair comparison we disregard the estimates obtained during the transient phase because they depend on the magnitude of the corresponding jump.

When the true  $d_m$  equals to 0.35[mm] for each loading phase the average bias of the estimate accounts for 14% of the true value, which is tolerable during the actual dredging. When the

## 7. SOLUTION TO THE HOPPER ESTIMATION PROBLEM

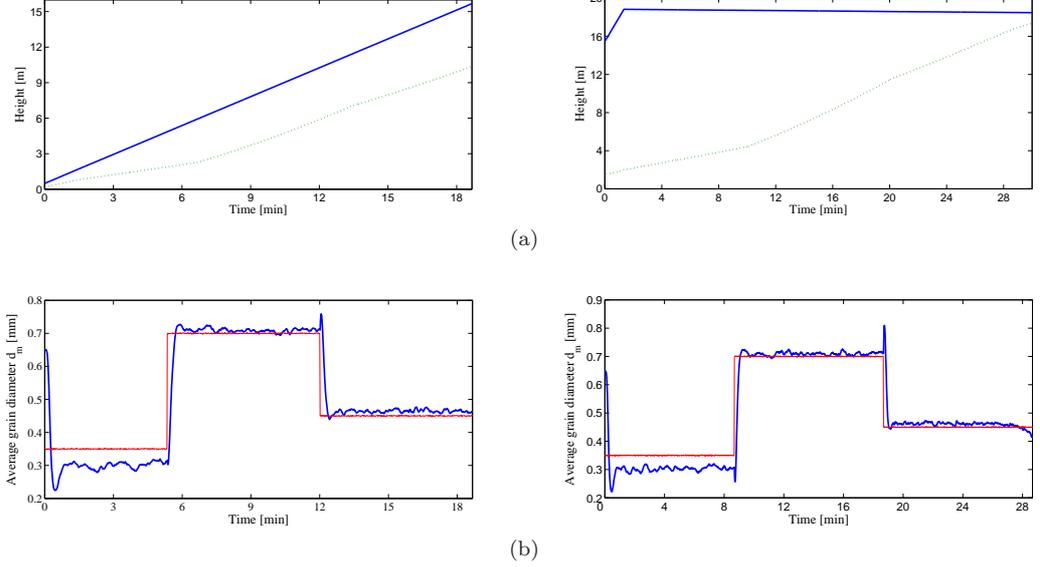


Figure 7.1: Above: simulated total height  $h_t$  (solid line) and sand bed height  $h_s$  (dashed line) during the No-Overflow phase (left) and during the constant-volume phase (right). Below: simulated average grain diameter  $d_m$  (thin solid line) and the estimated average grain diameter  $\hat{d}_m$  (thick solid line) during the No-Overflow phase (left) and during the Constant-Volume phase (right).

Table 7.2: Residuals obtained by the ROPF

true $d_m$	Res ( $\hat{d}_m$ ) for the No-Overflow		Res ( $\hat{d}_m$ ) for the Constant-Volume	
	mean [mm]	std [mm]	mean [mm]	std [mm]
0.35	-0.049	0.0095	-0.047	0.0077
0.7	0.01	0.0064	0.009	0.0066
0.45	0.014	0.0057	0.009	0.009

true signal is set to 0.7[mm] and 0.45[mm] for both loading phases, the average bias of the estimate is smaller than 3% of the true value, which is an excellent result from a practical point of view.

The results hereby presented were obtained by a filter that assumes the complete knowledge of the mixture density  $\rho_m$ . This assumption is valid for the Constant-Volume and Constant-Tonnage loading phases when the outgoing density  $\rho_o$  can be estimated [Lendek et al., 2008] and substituted for  $\rho_m$  [Braaksma, 2008] i.e.,

$$\rho_{m,k}(\mathbf{x}_k, \mathbf{u}_k) = \rho_o. \quad (7.21)$$

When the sand bed height  $h_s$  approaches the total height of the mixture  $h_t$ , the performance of the filter decreases, which can be observed in Figure 7.1b (last minutes of the Constant-Volume phase). This is due to the growing influence of the erosion factor  $f_e$  (7.4), which results in highly nonlinear behavior of the system that is intractable for the ROPF. The same applies for the whole Constant-Tonnage loading phase which is characterized by high erosion.

In current state-of-the-art dredgers, during the No-Overflow loading phase there is no ac-

## 7.2. Overflow Loading Phases with Weak Erosion: the Reduced-Order Particle Filter

curate measurements of the mixture density  $\rho_m$ . Therefore, in the next section we investigate the performance of the ROPF for which the  $\rho_m$  is unknown and has to be estimated online.

### Results for Scenario B ( $\rho_m$ is not known)

Now we investigate the situation when the mixture density  $\rho_m$  is unknown. This is the case during the No-Overflow loading phase when we cannot substitute the outgoing density  $\rho_o$  for the  $\rho_m$ .

Therefore, instead of one global and correct input  $\rho_{m,k}$ , each particle  $\mathbf{x}_k^i$  has to use a local approximation of the mixture density  $\rho_{m,k}^i$ . By (7.12) such a local approximation is given by:

$$\rho_{m,k}^i(\mathbf{x}_k^i, \mathbf{u}_k) = \frac{u_{1,k} - Ax_{2,k}^i \rho_s(x_{3,k}^i)}{Au_{3,k} - Ax_{2,k}^i}. \quad (7.22)$$

We have applied the ROPF to the simulations obtained from the model (7.1) of the sedimentation process during the No-Overflow loading phase. The filter is tuned with the parameters given in Table 7.1 the only difference being that we test two distinct values of the standard deviation  $\sigma_d = 0.1$  [mm] and  $\sigma_d = 0.05$  [mm] of the process noise  $e_d$ . This is to determine how sensitive the filter is with respect to this parameter. The results of the simulations are presented in Figure 7.2 where the estimates of the average grain diameter  $d_m$  and the mixture density  $\rho_m$  for the two choices of the parameter  $\sigma_d$  are presented.

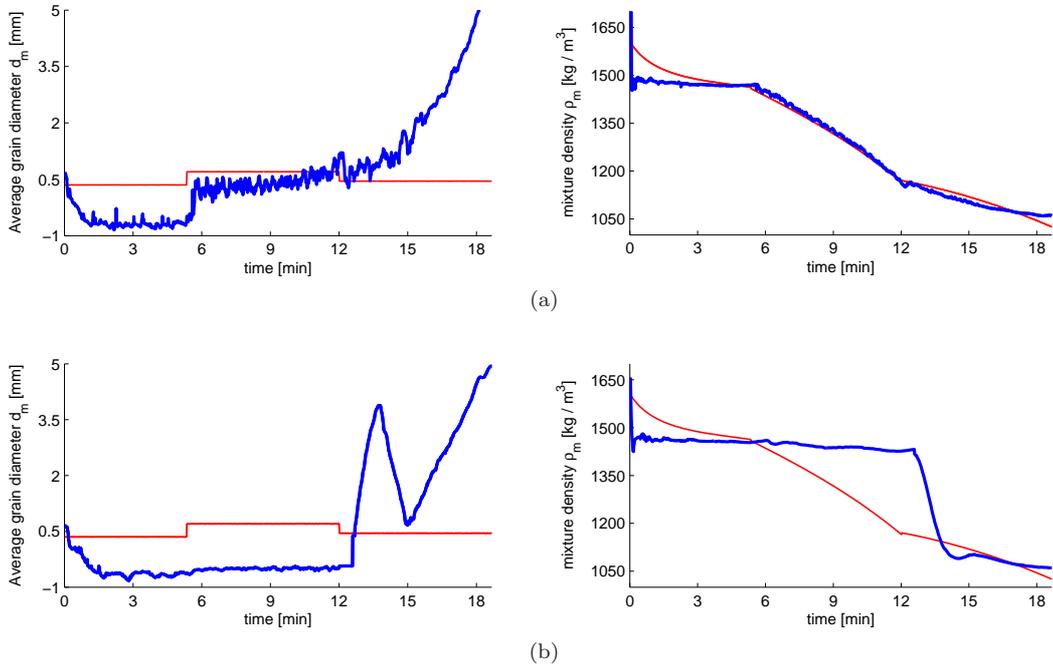


Figure 7.2: Left: simulated (thin solid line) and estimated (thick solid line) average grain diameter  $d_m$  (left) and mixture density  $\rho_m$  (right) for  $\sigma_d = 0.1$  [mm]. Right: simulated (thin solid line) and estimated (thick solid line) average grain diameter  $d_m$  (left) and mixture density  $\rho_m$  (right) for  $\sigma_d = 0.05$  [mm]

## 7. SOLUTION TO THE HOPPER ESTIMATION PROBLEM

---

The results of the simulations suggest that if the mixture density  $\rho_m$  is unknown the ROPF fails to estimate the average grain diameter  $d_m$ . Furthermore, it produces very poor estimates of the mixture density  $\rho_m$  which are very sensitive to the standard deviation  $\sigma_d$  of the process noise  $e_d$ . The high inaccuracy of the estimates that can be observed in Figure 7.2 makes it pointless to compute the residuals of the estimates  $\text{Res}(\hat{d}_m)$  or  $\text{Res}(\hat{\rho}_m)$ . Thus, the analysis of the residuals is not performed in this section.

### 7.2.3 Discussion

On board of the real TSHD, the measurements are taken with high frequency. However, the MPC designed for the TSHD [Braaksma, 2008] computes the optimal action in approximately 60 [s]. The proposed ROPF computes a single-step estimate in less than 0.25 [s] when executed in Matlab 7 on a PC with an Intel Core 2 Duo E6550 2.33 GHz CPU with 3 GB RAM. Thus, from the computational perspective, the filter can be integrated into the controller.

The simulation results suggest that the ROPF, using  $\rho_o$  for  $\rho_m$ , provides an accurate estimate of the average grain diameter  $d_m$  during the Constant-Volume phase as long as the erosion factor  $f_e$  is of small influence.

During the No-Overflow phase the ROPF produces an accurate estimate of  $d_m$  under the (hypothetical) assumption that the mixture density  $\rho_m$  is a known input to the system. In real dredging this assumption is not valid, thus the variable  $\rho_m$  needs to be estimated online. A properly tuned ROPF accurately estimates  $\rho_m$ , however, it fails to properly estimate the average grain diameter  $d_m$ . Moreover, it is very sensitive to the tuning of  $\sigma_d$ , which, if chosen incorrectly, causes the estimates of  $\rho_m$  to diverge.

To conclude, the ROPF is the recommended estimator of the the average grain diameter  $d_m$  after the overflow starts (Constant-Volume and Constant Tonnage loading phases) but before the erosion phenomenon dominates the dynamics of the process. Before the overflow commences (No-Overflow loading phase) the ROPF produces poor estimates of  $d_m$ .

## 7.3 No-Overflow Loading Phase: the Feedback Particle Filter

As we concluded in Section 7.2 the ROPF delivers satisfactory estimates only if specific conditions are satisfied. Namely:

- I. if the mixture density  $\rho_m$  is known, which is the case during the overflow mode, and
- II. if the influence of erosion is negligible (the first phase of the overflow mode).

In this section we provide solutions to the estimation problem for the No-Overflow loading phase during which the mixture density  $\rho_m$  is unknown and needs to be estimated. Furthermore, we consider a continuous-time filtering problem rather than the discrete-time problem discussed in Section 7.2. The objective is to estimate online the following states of the system:

- I. the average grain diameter  $d_m$ ,
- II. the mass of the sand bed  $m_s$ ,
- III. the density of the mixture  $\rho_m$ ,
- IV. the sand bed height  $h_s$ .

Note that, from the controller’s perspective, having accurate estimates of the average grain diameter  $d_m$  and the mass of the sand bed  $m_s$  is more important than having estimates of the remaining states.

We investigate two different algorithms that aim to achieve these goals. The first one is based on the classical Bootstrap Particle Filter (BPF) approach discussed in Chapter 3, the second is based on the recently developed Feedback Particle Filter (FPF) method, which has also been discussed in Chapter 3. For the continuous-time problem with the aforementioned objectives the reduced-order model (7.10) is insufficient. Instead, these two algorithms need to be based on the complete sedimentation model described in Chapter 3.

### 7.3.1 Derivation of the Stochastic System

In this section we formulate the continuous-time estimation problem based on the complete sedimentation model (7.1). The continuous-time formulation has two purposes: it allows discussing the estimation problem abstracting from the discretization errors, and it makes the formulation of the problem consistent with the formulation of the FPF of Chapter 3.

We assume that during the loading process four variables are observed with noises. These are: the height of the sand bed  $h_s$ , the total height of the mixture in the hopper  $h_t$ , the total mass of the mixture in the hopper  $m_t$ , and the incoming flow rate  $Q_i$ . Note that in general, the incoming flow  $Q_i$  can be considered as a known stochastic input variable, i.e., a known variable corrupted by noise. Therefore, in this section we consider  $Q_i$  as a measured state variable. The measurements are assumed to be corrupted by zero-mean, time-invariant Gaussian noises  $e_t^o$ ,  $e_s^o$ ,  $e_{m_t}^o$  and  $e_q^o$  with standard deviations  $\sigma_t^o$ ,  $\sigma_s^o$ ,  $\sigma_{m_t}^o$ , and  $\sigma_q^o$ , respectively. Also, we assume that the density of the incoming flow  $\rho_i$  is a known input signal.

From the conservation laws (7.1) we derive the stochastic dynamical system with six state variables: the sand bed mass  $m_s$ , the sand bed height  $h_s$ , the average grain diameter of the excavated soil  $d_m$ , the total height  $h_t$ , the total mass  $m_t$ , the incoming flow rate  $Q_i$ . Three state variables,  $h_t$ ,  $m_t$ , and  $Q_i$  have purely deterministic dynamics. Three remaining variables,  $m_s$ ,  $h_s$ , and  $d_m$  have the deterministic dynamics corrupted with stochastic zero-mean Gaussian noises  $e_{m_s}$ ,  $e_s$ , and  $e_d$  with standard deviations  $\sigma_{m_s}$ ,  $\sigma_s$ , and  $\sigma_d$ , respectively. The dynamics of the variables  $m_s$ ,  $h_s$ ,  $h_t$ ,  $m_t$  are derived directly from (7.1). For the purpose of simulations we assume the incoming flow  $Q_i$  to be constant, hence in our model

$$dQ_i(t) = 0dt. \tag{7.23}$$

Since we are interested only in the No-Overflow phase, the erosion does not influence the settling rate  $Q_s$  which then is modeled only as a function of four parameters: the average grain diameter  $d_m$ , the total mass in the hopper  $m_t$ , the sand bed height  $h_s$  and the total height of the mixture in the hopper  $h_t$  [Braaksma et al., 2007b]:

$$Q_s(d_m, m_t, h_s, h_t) = f_s(d_m, m_t, h_s, h_t), \tag{7.24}$$

where  $f_s$  is defined by (2.24).

The state, input and output vectors of the system are given by:

$$\mathbf{x} = \begin{pmatrix} m_s \\ h_s \\ d_m \\ h_t \\ m_t \\ Q_i \end{pmatrix}, \quad u = \rho_i, \quad \mathbf{y} = \begin{pmatrix} h_s \\ h_t \\ m_t \\ Q_i \end{pmatrix}.$$

## 7. SOLUTION TO THE HOPPER ESTIMATION PROBLEM

---

Then, the sedimentation dynamics are given by the stochastic dynamical system:

$$dx_1(t) = Q_s(\mathbf{x}_{2:5}(t)) \rho_s(x_3(t)) dt + de_{ms}(t), \quad (7.25a)$$

$$dx_2(t) = \frac{Q_s(\mathbf{x}_{2:5}(t))}{A} dt + de_s(t), \quad (7.25b)$$

$$dx_3(t) = 0dt + de_d(t), \quad (7.25c)$$

$$dx_4(t) = x_6(t)dt, \quad (7.25d)$$

$$dx_5(t) = u_1(t)x_6(t)dt, \quad (7.25e)$$

$$dx_6(t) = 0dt, \quad (7.25f)$$

$$dy_1(t) = x_2(t)dt + de_s^o(t), \quad (7.26a)$$

$$dy_2(t) = x_4(t)dt + de_t^o(t), \quad (7.26b)$$

$$dy_3(t) = x_5(t)dt + de_{mt}^o(t), \quad (7.26c)$$

$$dy_4(t) = x_6(t)dt + de_q^o(t). \quad (7.26d)$$

### 7.3.2 Numerical Simulations

To simulate the complete sedimentation process we discretize the continuous-time system (7.25)–(7.26) using the Euler discretization method. The sampling time is set to  $T_s = 0.05$ [s].

#### Setting

We consider two distinct simulation scenarios:

- I. Scenario 1: the average grain diameter  $d_m$  changes continuously notation from 0.4[mm] to 0.2[mm],
- II. Scenario 2: the average grain diameter  $d_m$  exhibits a step change in value from 0.4[mm] to 0.2[mm].

Scenario 1 describes a situation where the in situ soil changes constantly, yet slowly, while the dredging continues. Scenario 2 corresponds to a dredging operation where the ship sails over two zones of notably different in situ soils. Both these situation frequently occur during dredging operations.

It has been shown that the system (7.25) is very weakly sensitive to the soil-dependent parameter  $\beta(d_m)$  [Braaksma, 2008]. Therefore, for simplicity, for the average grain diameter in range  $d_m \in [0.20.4]$  we approximate  $\beta(d_m) \approx 3$ .

The hopper used for the simulations is of a rectangular parallelepiped form with the base area  $A = 600$ [m<sup>2</sup>]. In both scenarios the values of the incoming flow  $Q_i$  and the density of the incoming mixture  $\rho_i$  are assumed to be constant through out the simulation and equal to 8[m<sup>3</sup>/s] and 1600[kg/m<sup>3</sup>] respectively. The simulations start from the values given in Table 7.3.

Table 7.3: Initial state values

Variable	$m_s$ [tons]	$h_s$ [m]	$d_m$ [mm]	$h_t$ [m]	$m_t$ [tons]	$Q_i$ [m <sup>3</sup> /s]
Value	460	0.4	0.4	0.5	550	8

### 7.3. No-Overflow Loading Phase: the Feedback Particle Filter

We have simulated the dredging Scenarios 1 and 2 and to the obtained data we applied the BPF and the FPF. To investigate the robustness of both filters with respect to the initial conditions we have introduced the initial offset of +0.05 in the uncertain soil-dependent parameter  $d_m$ . The initial values of the remaining states are set to match the true values of the states given in Table 7.3. The parameters of the BPF and the FPF are presented in Table 7.4 and 7.4, respectively.

Table 7.4: Parameters of the BPF

Number of particles	Resampling threshold	Initial offset in $d_m$ [mm]	Standard deviations in process noises			Standard deviations in observation noises			
			$\sigma_{ms}$ [tons]	$\sigma_s$ [m]	$\sigma_d$ [mm]	$\sigma_t^o$ [m]	$\sigma_s^o$ [m]	$\sigma_{mt}^o$ [tons]	$\sigma_q^o$ [m <sup>3</sup> /s]
1000	500	+0.05	1	0.001	0.005 – 0.015	0.05	0.1	1	1

Table 7.5: Parameters of the FPF

Number of particles	Resampling threshold	Initial offset in $d_m$ [mm]	Standard deviations in process noises			Standard deviations in observation noises			
			$\sigma_{ms}$ [tons]	$\sigma_s$ [m]	$\sigma_d$ [mm]	$\sigma_t^o$ [m]	$\sigma_s^o$ [m]	$\sigma_{mt}^o$ [tons]	$\sigma_q^o$ [m <sup>3</sup> /s]
50	None	+0.05	1	0.001	0.005 – 0.015	0.05	0.1	1	1

## Results

The results of the estimations obtained for Scenario 1 and for Scenario 2 are reported in Figures 7.3 and 7.5, and in Figures 7.7 and 7.9, respectively. Figure 7.3 and Figure 7.7 show the results obtained by the BPF and the FPF with  $\sigma_d = 0.005$ [mm] whereas Figure 7.5 and Figure 7.9 show the results obtained by the BPF and the FPF with  $\sigma_d = 0.015$ [mm].

Figures 7.4, 7.6, 7.8, and 7.10 present the corresponding *Sample Standard Deviations* (SSDs) of the estimates  $\hat{x}(t)$ , which are defined by:

$$SSD(\hat{x}(t)) := \sqrt{\sum_{n=1}^N \left( \omega_n(t) \hat{x}_n(t) - \sum_{k=1}^N \omega_k(t) \hat{x}_k(t) \right)^2}, \quad (7.27)$$

where  $N$  is the number of particles  $\hat{x}_n(t)$  and corresponding weights  $\omega_n(t)$  at time  $t$ .

From Figures 7.4, 7.6, 7.8, and 7.10 one can observe that the SSDs obtained by the BPF are larger than the SSDs obtained by the FPF. For a fixed parameter  $\sigma_d$ , the limiting SSDs of the FPF are similar for Scenarios 1 and 2. The same holds for the SSDs of the BPF, except for the SSD of the estimate  $\hat{m}_s$  that, for  $\sigma_d = 0.015$ , varies significantly between Scenario 1 and Scenario 2.

We see that in both simulation scenarios the estimates of the sand bed height  $h_s$  and the mixture density  $\rho_m$  produced by the BPF and the FPF are very accurate. The estimates of the average grain diameter  $d_m$  and the sand bed mass  $m_s$  obtained by the BPF are relatively noisy and possibly divergent when the filter is inappropriately tuned (see Figures 7.5c–7.5d and 7.9c–7.9d). On the other hand the simulation results suggest that the FPF is robust with respect to the uncertain parameter of the random walk, i.e., regardless of  $\sigma_d$  the estimates eventually converge to the same values. Nevertheless, the convergence rate of the filter can be improved by appropriate tuning (compare Figures 7.3c–7.3d with Figures 7.5c–7.5d and Figures 7.7c–7.7d with Figures 7.9c–7.9d).

## 7. SOLUTION TO THE HOPPER ESTIMATION PROBLEM

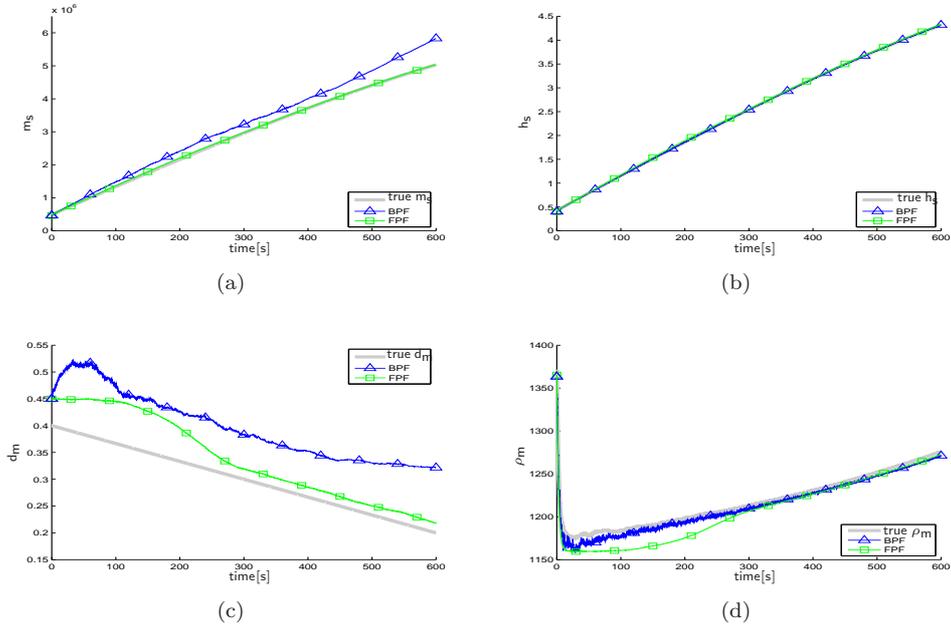


Figure 7.3: Comparison of the estimates produced by the PF (blue triangles) and the FPF (green squares) with the true state of the system (thick grey) for Scenario 1. The state noise, corresponding to the average grain diameter  $d_m$ , is set to  $\sigma_d = 0.005[\text{mm}]$ .

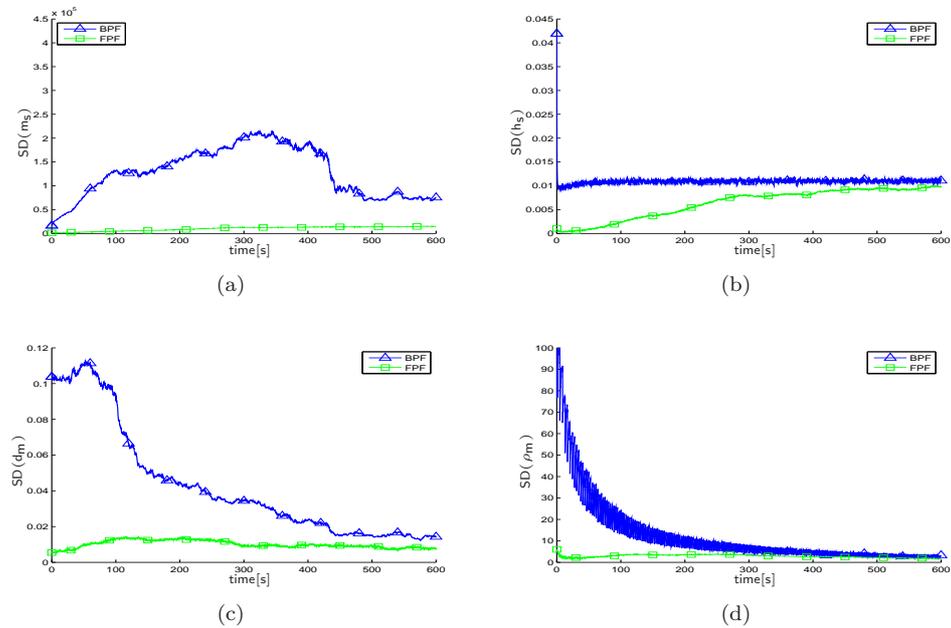


Figure 7.4: Comparison of the sample standard deviations obtained by the PF (blue triangles) and the FPF (green squares) for Scenario 1. The state noise, corresponding to the average grain diameter  $d_m$ , is set to  $\sigma_d = 0.005[\text{mm}]$ .

### 7.3. No-Overflow Loading Phase: the Feedback Particle Filter

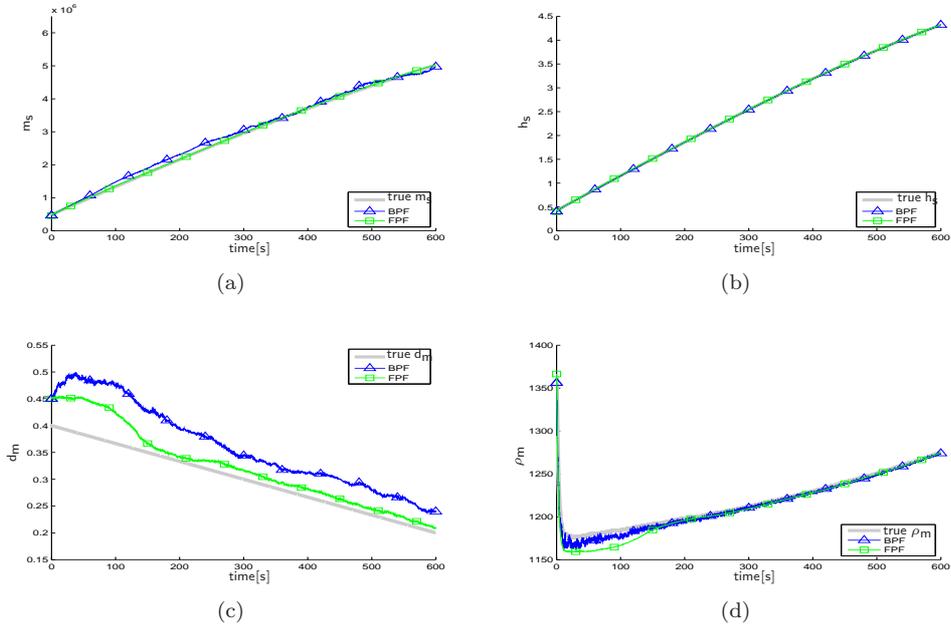


Figure 7.5: Comparison of the estimates produced by the PF (blue triangles) and the FPF (green squares) with the true state of the system (thick grey) for Scenario 1. The state noise, corresponding to the average grain diameter  $d_m$ , is set to  $\sigma_d = 0.015[\text{mm}]$ .

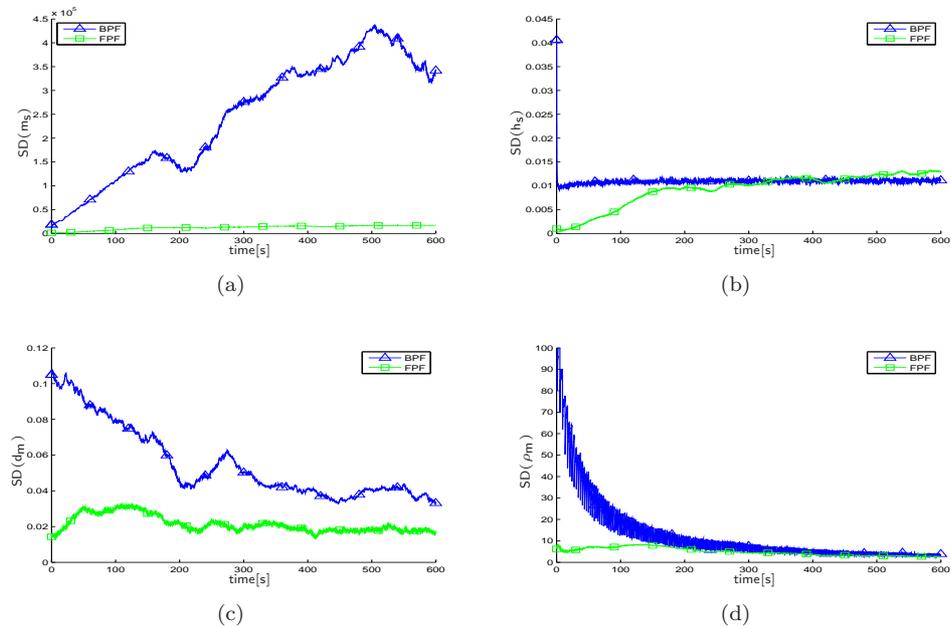


Figure 7.6: Comparison of the sample standard deviations obtained by the PF (blue triangles) and the FPF (green squares) for Scenario 1. The state noise, corresponding to the average grain diameter  $d_m$ , is set to  $\sigma_d = 0.015[\text{mm}]$  (for the PF).

## 7. SOLUTION TO THE HOPPER ESTIMATION PROBLEM

---

The quantitative measure of the estimation accuracy is provided by the MSE of the estimate  $\hat{x}$ , which in the continuous-time problem is defined by:

$$MSE(\hat{x}) := \frac{1}{T} \int_0^T (\hat{x}(t) - x(t))^2 dt, \quad (7.28)$$

where  $T$  is the total time of the simulation (in our case  $T = 600$  [s]). In other words, the MSE is the weighted  $L^2$  distance between the estimate  $\hat{x}$  and the true signal  $x$ . The MSEs of the estimates obtained by the BPF and the FPF for the Scenario 1 and Scenario 2 are compared in Table 7.6 and Table 7.7, respectively.

Table 7.6: Mean Squared Errors: Scenario 1

	$\sigma_d = 0.005$				$\sigma_d = 0.015$			
	$m_s$	$h_s$	$d_m$	$\rho_m$	$m_s$	$h_s$	$d_m$	$\rho_m$
PF	$1.171 \times 10^{11}$	$4.835 \times 10^{-6}$	0.0097	31.278	$1.0585 \times 10^{10}$	$2.2911 \times 10^{-6}$	0.0043	18.9846
FPF	$3.973 \times 10^8$	$3.708 \times 10^{-4}$	0.0021	221.0911	$1.8727 \times 10^8$	$4.4930 \times 10^{-5}$	0.0009	77.7335

Table 7.7: Mean Squared Errors: Scenario 2

	$\sigma_d = 0.005$				$\sigma_d = 0.015$			
	$m_s$	$h_s$	$d_m$	$\rho_m$	$m_s$	$h_s$	$d_m$	$\rho_m$
PF	$5.031 \times 10^{11}$	$1.247 \times 10^{-5}$	0.0153	45.875	$3.3555 \times 10^{11}$	$7.5463 \times 10^{-6}$	0.0113	34.3019
FPF	$2.196 \times 10^9$	0.0015	0.0044	426.1623	$2.4386 \times 10^8$	$2.17 \times 10^{-4}$	0.0022	120.95

It can be seen that the MSEs of the FPF-estimates of the sand bed mass  $m_s$  and the average grain diameter  $d_m$  are significantly smaller than the MSEs of the PF-estimates. On the other hand, the MSEs of the FPF-estimates of the sand bed height  $h_s$  and the density of the mixture  $\rho_m$  are larger than the MSEs of the PF-estimates. Thus, the comparison of the filters performance is inconclusive if they were judged solely by the resulting MSEs. It is only after the investigation of Figures 7.3, 7.5, 7.7, and 7.9 when we realize the character of the errors. We observe that large MSEs of the PF-estimates of the sand bed mass  $m_s$  and the average grain diameter  $d_m$  come from the persistent mismatch between the PF-estimates and the true states of the system, where the FPF-estimates are asymptotically unbiased. Per contra, large MSEs of the FPF-estimates of the sand bed height  $h_s$  and the density of the mixture  $\rho_m$  come from the transient state of the filter caused either by the initial offset (Figures 7.3b, 7.3d, 7.5b, 7.5d, 7.7b, 7.7d and 7.9b, 7.9d) or by the step-change in the system (Figures 7.7b, 7.7d and 7.9b, 7.9d). The steady-state errors of the FPF-estimates of the sand bed height  $h_s$  and the density of the mixture  $\rho_m$  converge to zero. Thus, judging by the simulation, we can safely conclude that error-wise the FPF outperforms the BPF applied to the system (7.25)–(7.26).

### 7.3.3 Discussion

In this section we have investigated the feasibility of the BPF and the FPF to provide the solution to the Hopper Estimation Problem for the No-Overflow loading phase without the knowledge of the mixture density  $\rho_m$ . Given the outcomes of the simulations we conclude that both filtering methods considered in this section are equal in producing asymptotically unbiased estimates of the mixture density  $\rho_m$  and the sand bed height  $h_s$ .

When it comes to the estimation of the average grain diameter  $d_m$  and sand bed mass  $m_s$  we observe a significant difference in the performance achieved by the BPF and the FPF. The FPF produces asymptotically unbiased estimates of  $d_m$  and  $m_s$  regardless of the choice of the tuning

### 7.3. No-Overflow Loading Phase: the Feedback Particle Filter

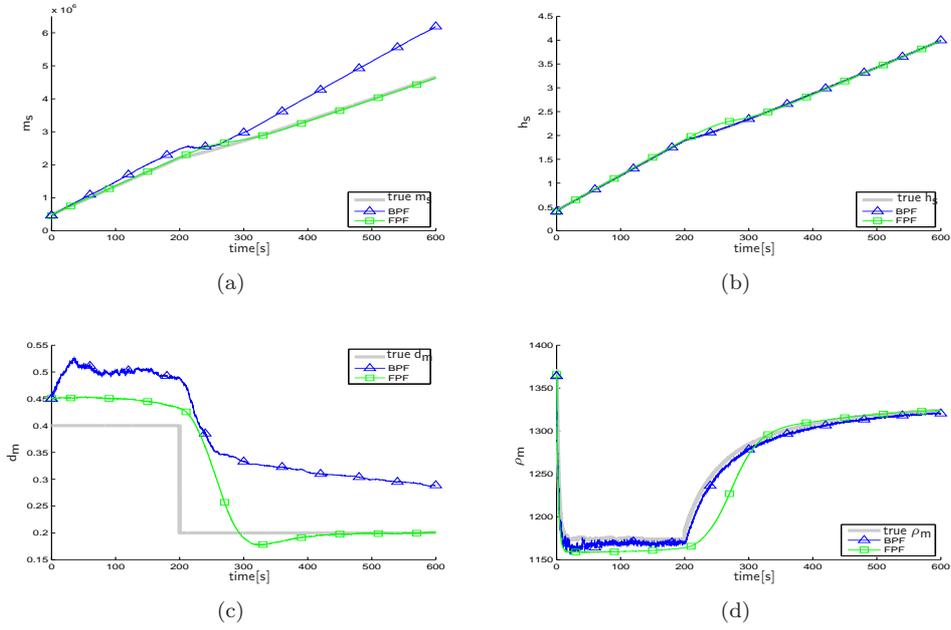


Figure 7.7: Comparison of the estimates produced by the PF (blue triangles) and the FPF (green squares) with the true state of the system (thick grey) for Scenario 2. The state noise, corresponding to the average grain diameter  $d_m$ , is set to  $\sigma_d = 0.005[\text{mm}]$ .

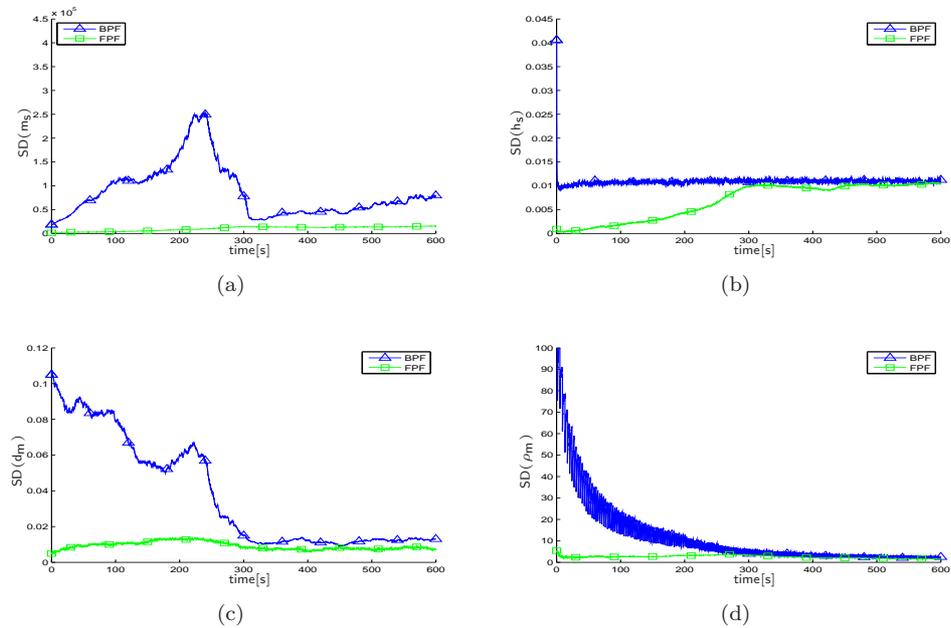


Figure 7.8: Comparison of the sample standard deviations obtained by the PF (blue triangles) and the FPF (green squares) for Scenario 2. The state noise, corresponding to the average grain diameter  $d_m$ , is set to  $\sigma_d = 0.005[\text{mm}]$ .

## 7. SOLUTION TO THE HOPPER ESTIMATION PROBLEM

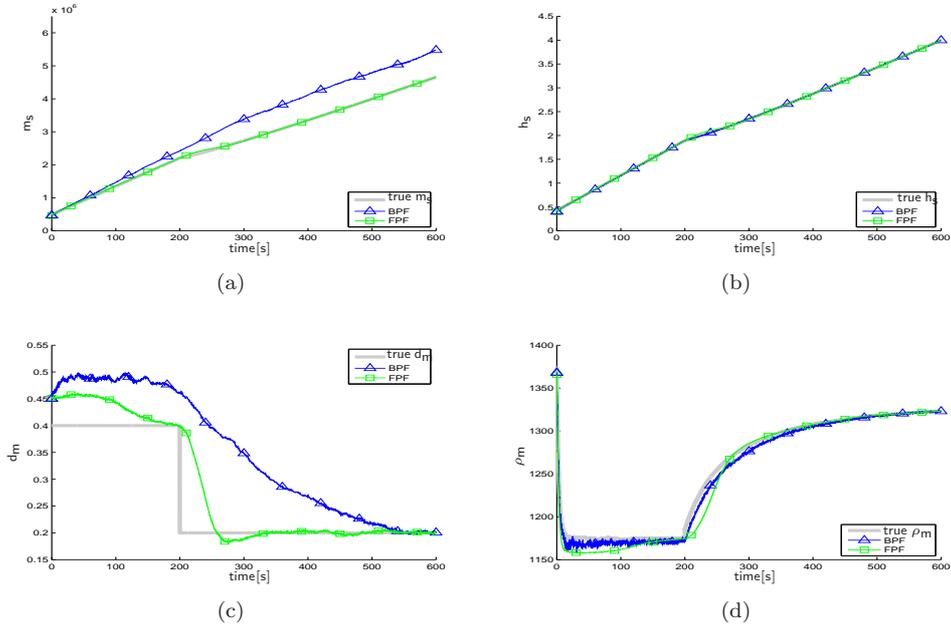


Figure 7.9: Comparison of the estimates produced by the PF (blue triangles) and the FPF (green squares) with the true state of the system (thick grey) for Scenario 2. The state noise, corresponding to the average grain diameter  $d_m$ , is set to  $\sigma_d = 0.015[\text{mm}]$ .

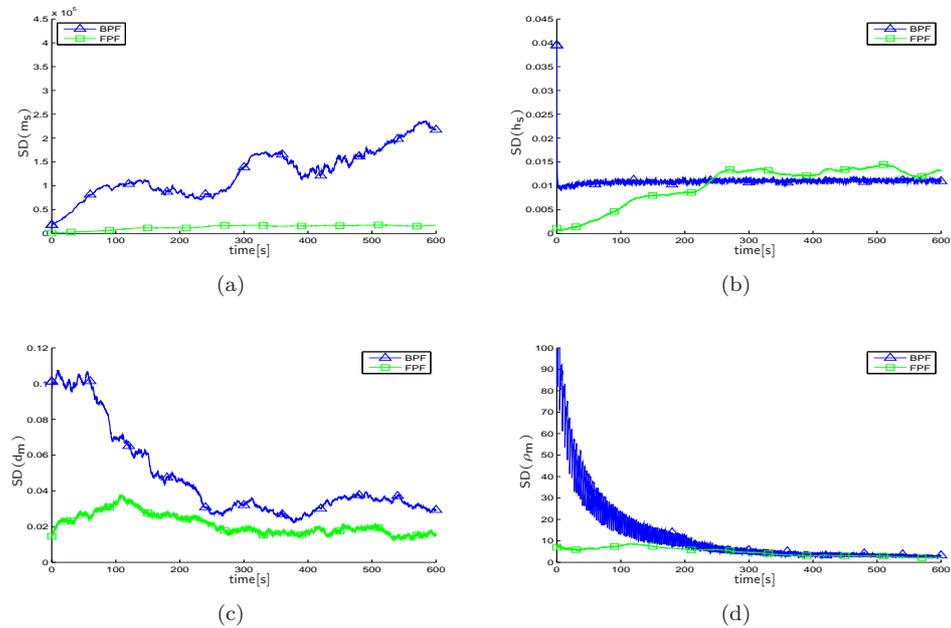


Figure 7.10: Comparison of the sample standard deviations obtained by the PF (blue triangles) and the FPF (green squares) for Scenario 2. The state noise, corresponding to the average grain diameter  $d_m$ , is set to  $\sigma_d = 0.015[\text{mm}]$ .

## 7.4. Overflow Loading Phases with Strong Erosion: the Improved Saturated Particle Filter

---

parameter  $\sigma_d$ . Per contra the PF requires very careful tuning in order to obtain asymptotically unbiased estimates of  $d_m$  and  $m_s$ . In other words the simulations showed that the FPF is robust and the BPF is highly sensitive with respect to the parameter  $\sigma_d$ .

The accuracy of the estimates produced by the BPF and the FPF is higher for higher values of the parameter  $\sigma_d$ . Moreover, the speed of convergence (settling time) of the FPF estimates of the average grain diameter  $d_m$  and the mixture density  $\rho_m$  is faster for higher values of  $\sigma_d$ .

The simulations also showed that, in general, the estimates obtained by the BPF are more, and in some cases significantly more, noisy than the estimates obtained by the FPF.

The simulations were performed in Python on a MAC OS X 10.6. with an Intel Core 2 Duo 2.66 GHz CPU with 4 GB RAM. Regarding the numerical properties of the compared filters it is important to notice that the FPF outperforms the BPF in terms of accuracy and precision while using 20 times less particle than the BPF does. This is a remarkable result as it suggests that the use of the FPF leads to a dramatic reduction of the computational effort.

Combining all these facts we conclude that during the No-Overflow loading phase the FPF provides an accurate solution to the Hopper Estimation Problem and it is recommended over the benchmark BPF.

## 7.4 Overflow Loading Phases with Strong Erosion: the Improved Saturated Particle Filter

In this section we analyze the second scenario in which the ROPF does not perform satisfactory, i.e., the mode with high erosion influence. The erosion phenomenon occurs in the second half of the Constant-Volume phase and throughout the Constant-Tonnage phase. In both these phases the value of the mixture density  $\rho_m$  is close to the value of the density of the outgoing mixture  $\rho_o$  which in turn can be accurately estimated from the available measurements [Lendek et al., 2008]. Thus, before the erosion influence becomes significant the ROPF introduced in Section 7.2 delivers accurate estimates of the average grain diameter  $d_m$ .

At some point the erosion influence becomes significant and the performance of the ROPF deteriorates. Since we do not know when exactly is the moment from which the erosion factor dominates the sedimentation dynamics we seek a filtering algorithm that would automatically switch from the ROPF to the estimator suited for the high erosion mode. Such an estimator is provided by the iSPF introduced in Chapter 5, which has its inherent mechanism of detection of changes in dynamics.

In what follows we show how to apply the iSPF, introduced in Chapter 6, to the sedimentation model. Furthermore, we analyze the feasibility of the iSPF to the Hopper Sedimentation Problem by comparing the estimates it produces with the estimates obtained by the ROPF.

### 7.4.1 Derivation of the Stochastic Dynamical System

In this section we show how the discrete-time dynamical sedimentation model (7.10) can be algebraically transformed into an equivalent form which is suitable for the application of the iSPF. The variables used in what follows match the variables introduced in Section 7.2.

## 7. SOLUTION TO THE HOPPER ESTIMATION PROBLEM

---

We start by observing that the derivative  $\dot{h}_s$  can be rewritten as:

$$\dot{h}_s = \max \left( 1 - \frac{Q_o^2}{(k_e(d_m)(h_t - h_s))^2}, 0 \right) \frac{f_s(d_m, \rho_m)}{A} \quad (7.29a)$$

$$= -\min \left( \frac{Q_o^2}{(k_e(d_m)(h_t - h_s))^2} - 1, 0 \right) \frac{f_s(d_m, \rho_m)}{A} \quad (7.29b)$$

$$= -\min \left( \frac{Q_o^2 f_s(d_m, \rho_m)}{(k_e(d_m)(h_t - h_s))^2} - f_s(d_m, \rho_m), 0 \right) \frac{1}{A}. \quad (7.29c)$$

Thus, the discrete-time model for the sand bed height  $h_s$  is given by:

$$h_{s,k+1} = h_{s,k} + T_s e_s - \frac{T_s}{A} \min \left( \frac{Q_{o,k}^2 f_s(d_{m,k}, \rho_{m,k})}{(k_e(d_{m,k})(h_{t,k} - h_{s,k}))^2} - f_s(d_{m,k}, \rho_{m,k}), 0 \right) \quad (7.30a)$$

$$= -\min \left( \frac{T_s}{A} \left( \frac{Q_{o,k}^2 f_s(d_{m,k}, \rho_{m,k})}{(k_e(d_{m,k})(h_{t,k} - h_{s,k}))^2} - f_s(d_{m,k}, \rho_{m,k}) - A e_s \right) - h_{s,k}, -h_{s,k} - T_s e_s \right). \quad (7.30b)$$

Then the state, input and output vectors of the system are given by:

$$\mathbf{x} = \begin{pmatrix} m_s \\ h_s \\ d_m \end{pmatrix}, \quad \mathbf{u} = \begin{pmatrix} m_t \\ Q_o \\ h_t \end{pmatrix}, \quad y = h_s.$$

and the final form of the state-space model is:

$$x_{1,k+1} = x_{1,k} + A \rho_s(x_{3,k})(x_{2,k} - x_{2,k-1}) + T_s A \rho_s(x_{3,k}) e_s + e_m, \quad (7.31a)$$

$$x_{2,k+1} = \min \left( \frac{T_s}{A} (S(\mathbf{x}_k, \mathbf{u}_k, e_t^o) - f_s(x_{2,k}, \rho_{m,k}(\mathbf{x}_k, \mathbf{u}_k)) - A e_s) - x_{2,k}, -x_{2,k} - T_s e_s \right), \quad (7.31b)$$

$$x_{3,k+1} = x_{3,k} + e_d, \quad (7.31c)$$

$$y_k = x_{2,k} + e_s^o. \quad (7.31d)$$

The function  $S$  is given by:

$$S(\mathbf{x}_k, \mathbf{u}_k, e_t^o) = \frac{u_{2,k}^2 f_s(x_{3,k}, \rho_{m,k}(\mathbf{x}_k, \mathbf{u}_k))}{(k_e(x_{3,k})(e_{t,k}^o + u_{3,k} - x_{2,k}))^2}, \quad (7.32)$$

and the variable  $\rho_{m,k}$  is given by:

$$\rho_{m,k}(\mathbf{x}_k, \mathbf{u}_k) = \frac{u_{1,k} - A x_{2,k} \rho_s(x_{3,k})}{A u_{3,k} - A x_{2,k}}. \quad (7.33)$$

Note that the erosion influences the system (7.31) only through the variable  $x_2$ . To obtain the estimates of the average grain diameter  $d_m$  that preserves good properties of the ROPF during the weak erosion phase and achieves better performance during the strong erosion phase we design a hybrid filter that applies the iSPF to the state  $x_2$  and the ROPF approach to the states  $x_1$  and  $x_3$ . Such a filter, called *Hybrid Saturated Particle Filter* (HSPF) is described in Algorithm 7.2.

## 7.4. Overflow Loading Phases with Strong Erosion: the Improved Saturated Particle Filter

---



---

### Algorithm 7.2 Hybrid Saturated Particle Filter

---

**Input:**  $\{(\mathbf{x}_k^i, \omega_k^i)\}_{i=1}^N, y_{k+1}, \mathbf{u}_k$   
**Output:**  $\{(\mathbf{x}_{k+1}^i, \omega_{k+1}^i)\}_{i=1}^N$   
**for**  $i = 1, 2, \dots, N$  **do**  
    **Prediction:**  
    Compute the predicted particles  $\mathbf{x}_{k+1|k}^i$ :  
    **if**  $j = 1, 3$  **then**  
        Draw the state particle  $x_{k+1|k,j}^i$  from the transition probability kernel  $K_k(x_{k+1,j}|\mathbf{x}_k, \mathbf{u}_k)$   
    **end if**  
    **if**  $j = 2$  **then**  
        Draw the state particle  $x_{k+1|k,j}^i$  from the importance kernel  $\tilde{K}_k(x_{k+1}| \mathbf{x}_k, \mathbf{u}_k, y_k)$  using Algorithm 5.2.  
    **end if**  
    **Update:**  
    Use the likelihood  $g_k(y_k|\mathbf{x}_k^i)$  to update the weights of the particles with formula (3.84)  
    **Resampling:**  
    Resample the particles with Algorithm 6.2  
**end for**

---

### 7.4.2 Algorithmic Properties

In order to successfully apply the iSPF to the sedimentation model first we need to tackle two numerical problems. These are:

- I. computing the saturation probability  $q_k^i$ ,
- II. drawing random samples from the saturated importance kernel  $\tilde{K}_k(x_{k+1}|\mathbf{x}_k, \mathbf{u}_k, y_k)$ .

In what follows we show that both these problems can be solved analytically without resorting to any approximations. This is possible by exploiting properties of the chi-square distribution.

Let us start from the observation that by (7.30) the probability of saturation  $q_k^i$  is defined by:

$$q_k^i = \mathbb{P} \left( \frac{(Q_{o,k}^i)^2}{(k_e (d_{m,k}^i) (h_{t,k}^i - h_{s,k}^i + e_t^o))^2} \geq 1 \right). \quad (7.34)$$

Note that (7.34) is the probability of the non-central inverse chi-square distribution  $\frac{1}{\chi^2}(\theta_{1,k}^i, \theta_{2,k}^i, \sigma_t^o)$  with parameters given by

$$\theta_{1,k}^i = \frac{k_e(d_{m,k}^i)}{Q_{o,k}^i}, \quad (7.35a)$$

$$\theta_{2,k}^i = h_{t,k}^i - h_{s,k}^i. \quad (7.35b)$$

Then, exploiting the relations between the inverse chi-square distribution and the chi-square distribution, the probability of saturation  $q_k^i$  is easily obtained by using the *Cumulative Density*

## 7. SOLUTION TO THE HOPPER ESTIMATION PROBLEM

---

*Function* (CDF) of the chi-square distribution. Namely:

$$q_k^i = CDF_{\chi^2\left(\frac{\theta_{2,k}^i}{\sigma_t^o}\right)}\left(\frac{1}{\left(\theta_{1,k}^i \sigma_t^o\right)^2}\right), \quad (7.36)$$

where  $\chi^2\left(\frac{\theta_{2,k}^i}{\sigma_t^o}\right)$  denotes the chi-square distribution with mean  $\theta_{2,k}^i$  and standard deviation  $\sigma_t^o$ .

The random sample from the saturated importance kernel is obtained by using the inverse CDF of the non-central inverse chi-square distribution  $\frac{1}{\chi^2}\left(\theta_{1,k}^i, \theta_{2,k}^i, \sigma_t^o\right)$ . It can be easily shown that for every  $u \in [0, 1]$ :

$$\left(CDF_{\frac{1}{\chi^2}\left(\theta_{1,k}^i, \theta_{2,k}^i, \sigma_t^o\right)}\right)^{-1}(u) = \frac{1}{\theta_{1,k}^i \sigma_t^o \left[CDF_{\chi^2\left(\frac{\theta_{2,k}^i}{\sigma_t^o}\right)}\right]^{-1}(1-u)}, \quad (7.37)$$

where  $\left[CDF_{\chi^2\left(\frac{\theta_{2,k}^i}{\sigma_t^o}\right)}\right]^{-1}$  is the inverse CDF of the chi-square distribution with mean  $\theta_{2,k}^i$  and standard deviation  $\sigma_t^o$ . The random sample from the saturated importance kernel is obtained in two steps:

- I. A sample  $u$  is drawn from the Uniform distribution  $\mathcal{U}([0, 1])$ , i.e.,

$$u \sim \mathcal{U}([0, 1]), \quad (7.38)$$

- II. The final sample  $x$  is computed by

$$x = \frac{1}{\theta_{1,k}^i \sigma_t^o \left[CDF_{\chi^2\left(\frac{\theta_{2,k}^i}{\sigma_t^o}\right)}\right]^{-1}(1-u)}. \quad (7.39)$$

The CDF and the inverse CDF of the chi-square distribution are available in standard numerical libraries, hence the probability of saturation  $q_k^i$  can be easily computed.

### 7.4.3 Numerical Simulations

#### Setting

We have simulated the sedimentation process in the high erosion regime to investigate the performance of the HSPF described in Algorithm 7.2. We compare the estimates obtained by the hybrid filter with the estimates produced by the ROPF.

In the simulations the average grain diameter  $d_m$  is constant throughout the loading process. We consider two distinct simulation scenarios, each corresponding to a different soil type:

- I. Scenario I: sedimentation of fine soil with  $d_m = 0.3[\text{mm}]$ ,
- II. Scenario II: sedimentation of coarse soil with  $d_m = 0.8[\text{mm}]$ .

## 7.4. Overflow Loading Phases with Strong Erosion: the Improved Saturated Particle Filter

For simplicity, the sampling time  $T_s$  is set to one. Throughout the simulations the incoming flow rate  $Q_i$  and the outgoing mixture density  $\rho_i$  are constant and set to  $Q_i = 4\text{m}^3/\text{s}$  and  $\rho_i = 1400\text{kg}/\text{m}^3$ . The overflow height  $h_{t,o}$  is also constant and is set to  $h_{t,o} = 7\text{ [m]}$ , which means the Constant-Volume loading phase.

The iSPF embedded into the HSPF uses the detection function  $\alpha$  defined by:

$$\alpha(x) = \begin{cases} \min(1, \log(50x + 1)) & \text{if } x > 0 \\ \max(-1, -\log(50x + 1)) & \text{otherwise} \end{cases} \quad (7.40)$$

The remaining parameters are the same for the iSPF and the ROPF. These are given in Table 7.8 (Scenario I) and in Table 7.9 (Scenario II).

Table 7.8: Parameters of the filters for Scenario I

Number of particles	Resampling threshold	Initial offset in $d_m$ [mm]	Standard deviations in process noises			Standard deviations in observation noises	
			$\sigma_m$ [tons]	$\sigma_d$ [mm]	$\sigma_s$ [m]	$\sigma_t^o$ [m]	$\sigma_s^o$ [m]
1000	500	+0.3	1	0.1	$10^{-10}$	0.1	0.025

Table 7.9: Parameters of the filters for Scenario II

Number of particles	Resampling threshold	Initial offset in $d_m$ [mm]	Standard deviations in process noises			Standard deviations in observation noises	
			$\sigma_m$ [tons]	$\sigma_d$ [mm]	$\sigma_s$ [m]	$\sigma_t^o$ [m]	$\sigma_s^o$ [m]
1000	500	+0.3	1	0.01	$10^{-10}$	0.1	0.025

### Results

The outcomes of the simulations are presented in Figure 7.11 and Figure 7.12. In Figure 7.11 we compare the estimates of the average grain diameter  $d_m$  produced by the ROPF and the HSPF. Figure 7.12 gives the corresponding SSDs of both filters.

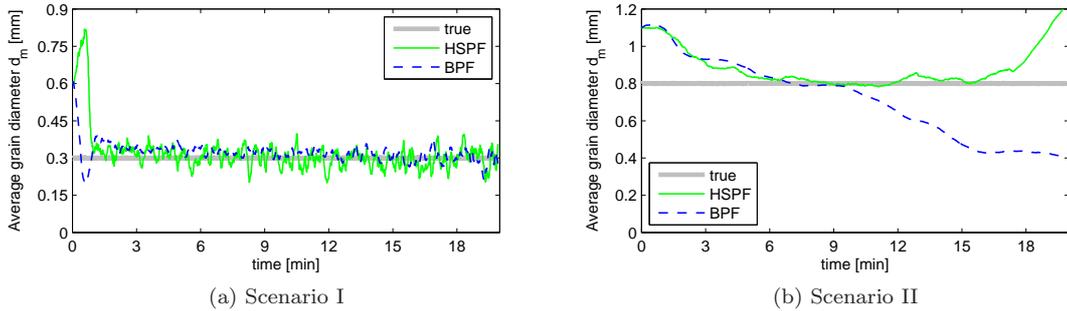


Figure 7.11: The HSPF estimation results (thin solid line) and the estimates obtained by the ROPF (dashed line) compared with the true value of  $d_m$  (thick solid line). The outcomes for Scenario I are presented on the left and those for Scenario II are presented on the right.

The results obtained for Scenario I suggest that if the excavated soil is fine then the accuracy of the HSPF is similar to the accuracy of the ROPF. The latter filter slightly outperforms

## 7. SOLUTION TO THE HOPPER ESTIMATION PROBLEM

the HSPF in terms of precision of the estimates. This can be observed in Figure 7.11a and Figure 7.12a and is confirmed by the analysis of residuals that are summarized in Table 7.10.

Table 7.10:  $d_m$ -residuals of the filters for Scenario I

Method	MSE	Standard deviation of residuals
HSPF	0.0083	0.0891
BPF	0.0021	0.0397

Table 7.11:  $d_m$ -residuals of the filters for Scenario II

Method	MSE	Standard deviation of residuals
HSPF	0.0183	0.1057
BPF	0.0529	0.2129

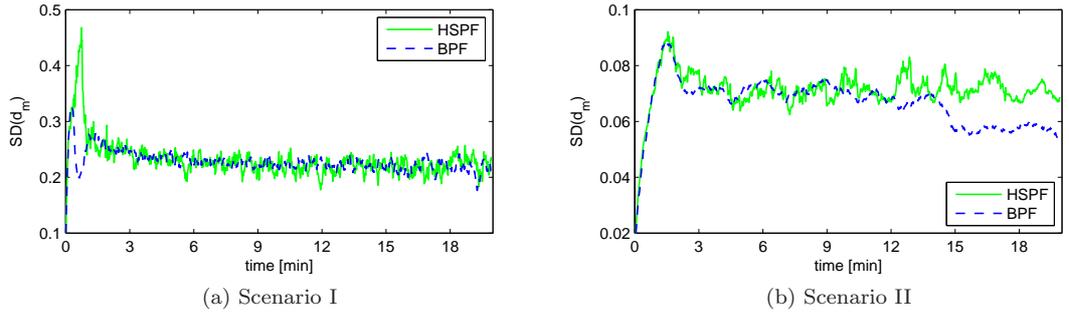


Figure 7.12: Standard deviations of the HSPF (thin solid line) and the ROPF (dashed line). The outcomes for Scenario I are presented on the left and those for Scenario II are presented on the right.

The results obtained for the coarse soil (Scenario II) suggest that for most of the phase the HSPF outperforms the ROPF in terms of accuracy of the estimates. However, at the end of the loading phase both filters become unstable which makes a fair comparison impossible. This can be observed in Figure 7.11b. The standard deviations of both filters are presented in Figure 7.12b and the residuals are given in Table 7.11.

The average time required by the HSPF to compute a single step estimate is 4.91 [s] whereas the average time required by the ROPF is significantly shorter and is equal to 0.12 [s]. This means that on average the HSPF is 41 times slower than the ROPF.

### 7.4.4 Discussion

In this section we have developed a novel algorithm, the HSPF, as an alternative to the ROPF to estimate the average grain diameter  $d_m$  during the loading phases that are heavily influenced by erosion. This is motivated by the fact that the ROPF does not perform satisfactorily in these final moments of the loading process.

The HSPF combines the iSPF approach introduced in Chapter 5 with the ROPF derived in Section 7.2. The simulations showed that the HSPF and the ROPF achieve similar accuracy if

the excavated soil is fine. However, if the excavated soil is coarse the HSPF is more accurate than the ROPF. Thus, in terms of accuracy of the estimates, the HSPF offers an improvement over the ROPF. On the down side the HSPF estimates have higher standard deviations than the estimates obtained by the ROPF. Thus, given both accuracy and precision, the recommended filtering strategy would be to use the ROPF for estimation if the excavated soil is fine and switch to the HSPF if the excavated soil is coarse. However, this strategy is feasible only when it can be detected if the soil is fine or coarse.

The simulations presented in this section were performed in Matlab 9 on a MAC OS X 10.6. with an Intel Core 2 Duo 2.66 GHz CPU with 4 GB RAM. In terms of computational load the HSPF is significantly more demanding than the ROPF. This is due to stochastic properties of the algorithm described in Section 7.4.2. The numerical effort can be reduced by use of approximate methods, e.g., approximate sampling described in Algorithm 7.1. However, introducing such approximations might decrease the accuracy of the filter. Hence any such trial needs to be carefully thought through.

## 7.5 Conclusions

In this chapter we have provided the solution to the Hopper Estimation Problem that was formulated in Chapter 2. We have investigated several estimation algorithms in order to find the best solution to each phase of the process. We have distinguished three such operational modes that appear naturally during the dredging operations. These are:

- I. Mode 1: No-Overflow loading phase,
- II. Mode 2: Overflow loading phases (Constant-Volume and Constant-Tonnage) without strong erosion influence,
- III. Mode 3: Overflow loading phases (Constant-Volume and Constant-Tonnage) with strong erosion influence.

For the estimator that solves the Hopper Estimation Problem for all three aforescribed modes we recommend a hybrid of three distinct filters:

- I. For Mode 1, as we concluded in Section 7.3, the FPF is the recommended estimation algorithm. The FPF provides very accurate estimates of the average grain diameter  $d_m$  as well as accurate estimates of the mixture density  $\rho_m$ . Furthermore, it comes with a low computational price as the FPF uses a very small number of particles.
- II. For Mode 2, as we concluded in Section 7.2, the ROPF, which assumes the knowledge of  $\rho_m$ , provides accurate estimates of  $d_m$ . This algorithm is also very fast thanks to simplified state model that it employs.
- III. For Mode 3, as we concluded in Section 7.4, the recommended solution to the estimation problem consists of applying the ROPF for the fine excavated soil and the HSPF for the coarse excavated soil. In this case the HSPF is numerically expensive which is a serious drawback of the method. Both the HSPF and the ROPF fail to estimate the average grain diameter  $d_m$  during the last minutes of the loading process. However, this is of low importance for the production process as this stage corresponds to the moment when the erosion gets so high that the process is terminated as it is not economical to continue the dredging operation.

## 7. SOLUTION TO THE HOPPER ESTIMATION PROBLEM

---

It is important to notice that the moment of switch between Mode 1 and Mode 2 is always known as we can observe when the overflow starts. Per contra the switch between Mode 2 and Mode 3 cannot be directly observed and has to be detected from data.

### Further Research

For the further research on how to improve the filtering algorithm used in the Hopper Estimation Problem we suggest to investigate the following two topics:

- I. Application of the FPF to Mode 2 and Mode 3. The FPF is a very recently developed algorithm whose full potential has yet to be determined. However, current results, both theoretical and applied, are very promising and suggest that in future the FPF can become a benchmark nonlinear filter. This is especially important for practitioners as the switching strategy described above can be difficult to implement into onboard decision system.
- II. Appropriate tuning of the iSPF embedded in the HSPF can possibly improve the performance of the HSPF. In particular in some cases the performance of the iSPF strongly depends on the choice of the detection function. For the Hopper Estimation Problem the detection function (7.40) was determined experimentally because the optimal choice of detection function has not been tackled yet.

# Chapter 8

## Conclusions

Research on a smart automatic controller of a Trailing Suction Hopper Dredger (TSHD) aims to improve the cost-efficiency of dredging operations. Due to the increasing demand for TSHDs in offshore projects, the automation methods have been extensively studied in the dredging community in recent years. Despite significant developments in the field, a fully automated efficient controller has yet to be constructed. One of the main obstacles is the identification of a number of uncertain soil-dependent parameters. The difficulty of such a task comes from several factors: from lack of sufficient sensors; from severely nonlinear models; from time-varying nature of the parameters, to name a few. In the research described in this thesis we focused on developing fast and efficient estimation methods for the most important soil-dependent parameters.

### 8.1 Summary

The parameters we consider in this thesis are: the horizontal cutting force coefficient  $k_{ch}$ , the ratio  $k_{vh}$  between the horizontal and vertical cutting forces, the in situ permeability  $k_{si}$  and the average grain diameter  $d_m$ . The first three, i.e.,  $k_{ch}$ ,  $k_{vh}$ , and  $k_{si}$  are associated with the Drag-Head model and  $d_m$  comes from the Hopper model. These two models describe two of the most important processes during the dredging: the excavation process and the sedimentation process. Both processes are discussed in detail in Chapter 2. Furthermore, in the same chapter, the corresponding estimation problems are formulated. In both cases the estimation goal is to retrieve the knowledge of the in-situ soil properties online from indirect measurements.

The highly uncertain and time-varying nature of the parameters of interest and nonlinear dynamics of the considered systems in each case make estimation a challenging undertaking. The nonlinear filtering methods that are suitable for these types of problems are reviewed in Chapter 3. These are divided into five types:

- I. Parametric Filters based on analytical approximations: Extended Kalman Filter (EKF), Iterated Extended Kalman Filter (IEKF).
- II. Parametric Filters based on statistical approximations: Unscented Extended Kalman (UKF), Gauss-Hermite Filter (GHF), Central Difference Filter (CDF).
- III. Parametric Filters based on Gaussian Sum approximations: Gaussian Sum Filter (GSF).

## 8. CONCLUSIONS

---

- IV. Nonparametric Filters based on the importance sampling approach: Bootstrap Particle Filter (BPF).
- V. Nonparametric Filters based on the mean-field control-oriented approach: Feedback Particle Filter (FPF).

The EKF and the IEKF are based on the Taylor series expansion of nonlinear functions which define the dynamics of the system. The UKF, the GHF and the CDF are based on statistical approximations of nonlinear transformations. The EKF, the IEKF, the UKF, the GHF, and the CDF approximate the predicted and the posterior densities as Gaussians whereas the GSF approximates them by Gaussian Sums, which are no longer Gaussian. Nonparametric filters do not assume any particular shape of the predicted and the posterior densities and thus are capable of approximating PDFs of any kind.

The filters discussed in Chapter 3 are used in Chapter 4 to solve the Drag-Head estimation problems. We distinguish two such problems: the Cutting Estimation Problem and the Cutting and Jetting Estimation Problem. The Cutting Estimation Problem is to estimate two soil-dependent parameters: the ratio  $k_{vh}$  between cutting forces and the horizontal cutting force coefficient  $k_{ch}$ . The Cutting and Jetting Estimation Problem is to estimate the horizontal cutting force coefficient  $k_{ch}$  and the in situ permeability  $k_{si}$ . The Cutting Estimation Problem applies for any cutting excavation tool whereas the Cutting and Jetting Estimation Problem is applicable only for tools equipped with cutting and jetting components. For each of the estimation problems we have tested several filtering methods in order to find the best solution.

Both estimation problems share a common feature of time-varying delay in the measurement of incoming density  $\rho_i$ . Handling such a delay is a problem independent of the estimation method and is discussed separately.

To solve the Hopper Estimation Problem that was formulated in Chapter 2, in Chapter 5 we have developed a novel nonparametric filtering method, the Saturated Particle Filter (SPF), that is tailored for this specific type of systems, namely, for Saturated Stochastic Dynamical Systems (SSDS) that are severely nonlinear systems often met in real life problems. The SSDS is a stochastic system with dynamics characterized by a constrained probability distribution exhibiting singularity on the boundary of the constraint region. Due to their complicated dynamical structure it is difficult to estimate the states or the parameters of the SSDSs by standard parametric methods. Our new method exploits the specific structure of the SSDS in order to design an importance sampling distribution that accounts for the most recent measurements in the prediction step of the filtering algorithm. This is possible through the use of a so-called detection function.

The asymptotic properties of the SPF are established in Chapter 6. Namely, we have given the conditions under which the SPF converges to the optimal theoretical filter. The convergence of our method is closely related to the appropriate resampling scheme. This led to the development of the improved Saturated Particle Filter (iSPF) by introducing a novel resampling algorithm.

The iSPF together with other nonparametric methods described in Chapter 3 are used in Chapter 7 to solve the Hopper Estimation Problem defined in Chapter 2. The objective of the Hopper Estimation Problem is to design an online estimator for the average grain diameter  $d_m$ . The sedimentation process is divided into three regimes that appear naturally during dredging operations. To find the most efficient filtering method we considered each of the modes separately. The final solution to the Hopper Estimation Problem is obtained by integrating the filters designed for separate modes into a global estimator.

## 8.2 Thesis Contributions

This thesis contributes to two fields: to the theory of nonlinear estimation methods and to the application of control systems in the dredging industry.

### 8.2.1 Theoretical Contributions

The main theoretical contribution of this thesis is the development of novel efficient estimation methods suitable for SSDSs. Chapters 5–6 are devoted to the description of these methods.

First, in Section 5.2 the nonparametric SPF is derived for one-dimensional SSDSs. The main idea behind the SPF is to combine the existing projection approach with a novel sampling method to detect the saturation moment as it occurs, and to force the particles to move towards the part of the state space which is close to the saturation region. Such sampling is obtained by means of a specially designed importance density function that makes use of both the measurement and the knowledge of the system constraints.

The SPF is improved in Section 6.4 where the iSPF is derived. The new algorithm is different from the standard SPF in two ways:

- I. a novel resampling procedure is used to discard certain low weighted particles,
- II. the detection function dynamically adapts to the states of the system.

The main advantage of the iSPF over the SPF comes from the resampling method that yields the former filter more accurate and computationally more efficient. Namely, the resampling reduces the computational load of the filter by preventing expensive numerical computations on the particles with negligible weights.

The dynamic nature of the detection function increases the flexibility of the filter making it applicable for essentially any type of one-dimensional SSDS. The influence of the detection function is visible already for a small number of particles. Furthermore, the results of the simulations suggest that the SPF is robust with respect to the choice of the detection function.

We have compared the performance of our method with the benchmark method, the Constrained Bootstrap Particle Filter (CBPF), in several simulation settings including such situations as model mismatch, large offset in the initial conditions, etc. The results of the simulation studies demonstrated that in each case our method outperforms the benchmark filter in terms of accuracy, speed of convergence and numerical efficiency. To achieve good performance, the CBPF required many particles (more than 1000), whereas our method is very accurate even when using few particles (10 particles).

The successful application of the SPF to simulated case studies encouraged us to extend the one-dimensional SPF to multidimensional SSDSs. This is done in Section 5.3 where we showed that such an extension requires an extra condition to be imposed, namely the constraints of the considered system need to be convex. Such an assumption allows for a proper definition of a multidimensional detection function, which is used to define a multidimensional analogue of the SPF, the Convex Saturated Particle Filter (CSPF).

We have proven that, under certain conditions, the finite-point PDFs obtained by the newly proposed method converge almost surely to the true posterior PDF. The sufficient conditions to establish such a convergence for the SPF are provided in Theorem 6.2 whereas the almost sure convergence of the iSPF is proven in Theorem 6.3. Thanks to the new resampling procedure utilized by the iSPF the assumptions of Theorem 6.3 are less restrictive than the assumptions of Theorem 6.2. In other words, the iSPF achieves a proper asymptotic behavior for a broader class of systems than the SPF does.

## 8. CONCLUSIONS

---

### 8.2.2 Practical Contributions

The main practical contributions of this thesis are:

- I. solutions to the Drag-Head Estimation Problems,
- II. solutions to the Hopper Estimation Problem.

The algorithms that we are proposing achieve high statistical performance while preserving low computational complexity which makes them suitable for online applications.

#### Solutions to the Drag-Head Estimation Problems

The Drag-Head Estimation Problems are solved in Chapter 4. The problem of time-varying delay in measurements and two estimation problems: the Cutting Estimation Problem and the Cutting and Jetting Estimation Problem are tackled independently. The algorithm connecting the delay problem with the estimation problem is described in Algorithm 4.2 and schematically depicted in Figure 4.3.

In case of the Drag-Head one can switch from one estimation problem to the other simply by turning on/off the jets. Because of that, it is possible to obtain the estimates of all three soil-dependent parameters associated with the Drag-Head by the following procedure:

- I. First, the excavation process is commenced with the jets turned off. In this mode it is possible to obtain an accurate estimate of  $k_{vh}$  by solving the Cutting Estimation Problem with any of the parametric filters discussed in Section 4.3.
- II. Next, the jets are turned on to increase the production rate. In this mode it is possible to solve the Cutting and Jetting Estimation Problem using the previously obtained estimate of  $k_{vh}$  as an input to the system.

We have tested four algorithms, the EKF, the UKF, the CDF, and the GSF to find the best solution to the Cutting Estimation Problem. The simulation results suggest that each of the discussed nonlinear filters provide a good solution to the Cutting Estimation Problem. In fact, all the filters produce estimates that are almost indistinguishable among each other in terms of MSE, overshoot, or convergence times, the CDF slightly outperforming the remaining methods. When the filters have to counteract a large uncertainty in the initial state of the system, which results in a large offset, the multi-term GSF outperforms the remaining filters. This is possible because the structure of the GS approximation of the state uncertainty allows the GSF to cover the uncertainty region with higher precision. The price of this is the higher computational complexity when compared to, e.g., a very simple and fast CDF. Thus, when the uncertainty in the initial states is small, i.e., we know the exact position of the system, the CDF is recommended over the GSF.

To determine the best solution to the Cutting and Jetting Estimation Problem we have investigated several approaches based on nonlinear nonparametric filtering methods. The numerical simulations showed that exploiting the correlation between the soil-dependent parameters the horizontal cutting force coefficient  $k_{ch}$  and the in situ permeability  $k_{si}$ , is of crucial importance for the development of an accurate estimator. The simulations suggest that the best results are obtained by the cascaded filter which first employs the PF to obtain an estimate of  $k_{si}$ ,  $k_{si}$  is further filtered by the SSI filter, and finally by the BF, which produces an estimate of  $k_{ch}$ .

### Solution to the Hopper Estimation Problem

The Hopper Estimation Problem is solved in Chapter 7. Several techniques are investigated in order to find a solution most feasible to each of the dredging operational modes

- I. Mode 1: No-Overflow loading phase,
- II. Mode 2: Overflow loading phases with weak erosion influence,
- III. Mode 3: Overflow loading phases with strong erosion influence.

The estimator that solves the Hopper Estimation Problem for all the above modes is a hybrid of three distinctive filters:

- I. For Mode 1, as has been shown in Section 7.3, the FPF outperforms the benchmark BPF. The FPF provides very accurate estimates of the average grain diameter  $d_m$  as well as accurate estimates of the mixture density  $\rho_m$ . Furthermore, the FPF achieves this using very few particles, which greatly reduces the numerical effort of the algorithm.
- II. For Mode 2, as was concluded in Section 7.2, the Reduced-Order PF, which assumes the knowledge of  $\rho_m$ , provides accurate estimates of  $d_m$ . Thanks to the simplified state model that the Reduced-Order PF uses, the computational cost of the algorithm is low.
- III. For Mode 3, as has been shown in Section 7.4, the recommended solution to the Hopper Estimation Problem is to apply the Reduced-Order PF when the excavated soil is fine and the Hybrid Saturated Particle Filter (HSPF) when the excavated soil is coarse. The improved performance of the HSPF comes with the price of higher numerical complexity of the method.

The moment of switch between Mode 1 and Mode 2 is always known because we can observe when the overflow starts. On the other hand the moment of switch between Mode 2 and Mode 3 cannot be directly observed and has to be detected from the available data.

## 8.3 Further research

There are several points that have not been discussed in details in this thesis. To further improve the results of this thesis we suggest to investigate the following topics.

### Research on the Saturated Particle Filter

- *Construction of the optimal detection function.* The influence of the detection function depends on the appropriate choice of the constant  $\epsilon > 0$  in Algorithm 6.3. On the other hand the choice of  $\epsilon > 0$  determines the bounds on the variance of the weights associated with particles. The smaller  $\epsilon$ , the stronger the influence of the detection function is, but, at the same time, the bigger is the upper bound on weights. The exact nature of this relation is a matter of the ongoing research.
- *Asymptotic properties of the CSPF.* The CSPF has been derived as the multidimensional extension of the SPF. Thus, it is our belief that the algorithm can be further improved by designing an appropriate resampling scheme that resembles the method developed for the iSPF. Furthermore, the asymptotic behavior of the CSPF has yet to be investigated e.g., by following the proofs of Theorem 6.2 and Theorem 6.3.

## 8. CONCLUSIONS

---

### Research on the Drag-Head Estimation Problems

Deeper investigation of the topics listed below can potentially lead to further improvements of the estimation methods discussed in this thesis:

- *Combining the Cutting Estimation Problem with the Cutting and Jetting Estimation Problem.* Separately, under the corresponding assumptions, each problem has been solved. We believe that combining solutions to both problems into a single algorithm would lead to the relaxation of the assumptions imposed on the filters. This could be achieved e.g., by the previously discussed strategy of switching between the Cutting mode and the Cutting and Jetting mode.
- *Uncertainty in input variables.* In our investigations we have assumed that the uncertainty in certain input variables is negligible. Relaxation of this assumptions might reflect reality better, leading to the development of algorithms that are of greater value for practitioners.
- *Optimal prior for the BF.* The overall performance of the cascaded filter used for the Cutting and Jetting Estimation Problem could be further improved by designing an optimal prior for the BF embedded into the cascaded filter.
- *Robustness of the cascaded filter with respect to correlation between the parameters  $k_{ch}$  and  $k_{si}$ .* The performance of the cascaded filter depends on the functional relation between the horizontal cutting force coefficient  $k_{ch}$  and the in situ permeability  $k_{si}$ . These are only rough empirical approximations of complex relations between soil-dependent parameters. Thus, to account for approximation errors, the cascaded filter should be robust with respect to these approximations.

### Research on the Hopper Estimation Problem

- *Applying the FPF to all phases of the sedimentation process.* The FPF is a very recently developed algorithm whose full potential has yet to be determined. However, current results, both theoretical and applied, are very promising and suggest that in the future the FPF can become a benchmark nonlinear filter. Therefore, the performance of the FPF should be tested on Mode 2 and Mode 3 of the sedimentation process.
- *Tuning of the HSPF.* Appropriate tuning of the iSPF embedded in the HSPF can possibly improve the performance of the filter. In particular in some cases the performance of the SPF strongly depends on the choice of the detection function. This is connected to the more general problem of finding an optimal detection function for the SPF.

## Appendix A

# Uncertainty Analysis via Monte Carlo Simulations

The cutting input  $u_c$  and the jetting input  $u_j$  are defined by:

$$u_c := \frac{r_{vc}}{x_2} F_{vc} (\sin(\alpha_{lt} + \alpha_v) + k_{vh} \cos(\alpha_{lt} + \alpha_v))^{-1} (h_z + 10)^{-1}, \quad (\text{A.1a})$$

$$u_j := C_{dh} p_j^{0.5} Q_{w,j} v_{sh}^{-1}, \quad (\text{A.1b})$$

where  $x_2$  and  $C_{dh}$  are parameters known from the specifications of the Drag-Head and  $k_{vh}$  defined in (2.7) is assumed to be known. All the remaining variables are described in Table 2.1.

The variables  $u_c$  and  $u_j$  are stochastic as they are corrupted by noises that come from other measured variables ( $p_j, Q_{w,j}, v_{sh}, r_{vc}, F_{vc}, \alpha_{lt}, \alpha_v, h_z$ ) with precisions given in Table 2.1. We analyze the distributions of variables  $u_c$  and  $u_j$  by Monte Carlo experiment designed for the typical values of the two variables taken from Tables A.1 and A.2. The values of  $u_c$  and  $u_j$  reported in Tables A.1 and A.2 are computed from the typical values of  $p_j, Q_{w,j}, v_{sh}, r_{vc}, F_{vc}, \alpha_{lt}, \alpha_v, h_z$  using (A.1). For each of these components we generate random samples from a zero-mean Gaussian with standard deviations derived from Table 2.1. Next, we propagate the variables, corrupted by these noises, through (A.1) in order to compare them with the true values of  $u_j$  and  $u_c$  (uncorrupted).

The simulations suggest that the noise associated with  $u_c$  is distributed normally with zero-mean, and that the noise associated with  $u_j$  is slightly skewed to the right, which suggests the presence of a bias in the signal. Several Monte Carlo experiments, each with  $10^6$  samples, are reported in Tables A.1 and A.2. Furthermore, three representative simulations are reported in Figures A.1–A.3.

As can be observed from Figures A.1a–A.3a the uncertainty in the cutting input  $u_c$  is very well approximated by a Gaussian with zero mean. This is confirmed by the Kolmogorov-Smirnov test for goodness of fit [Feller, 1948; Massey, 1951] comparing the empirical samples with a Gaussian with sample mean and sample variance. The significance level of the test is

Table A.1: Results of Monte Carlo experiments for different values of the cutting input  $u_c$ . P-value refers to the Kolmogorov-Smirnov test for goodness of fit to a Gaussian with parameters given by sampled mean and sampled std.

true value of $u_c$	sample mean	sample std	p value
4160.5	4160.5	119.6	0.974
5163	5163	120	0.924
5369.5	5369.5	119.9	0.754
5387.7	5387.7	119.7	0.946
5892.1	5892.3	120	0.975
7527.7	7527.7	120.4	0.581
7529.2	7529.3	120.8	0.954
7589.1	7589.3	120.3	0.909
7613.6	7613.6	120.4	0.858
7855.4	7855.3	120.4	0.593
8122.4	8122.5	120.1	0.634
8227	8227.1	120.8	0.968
8666.7	8666.8	120.7	0.943
8729.7	8729.8	120.4	0.941
9050.3	9050.2	120.9	0.972
9354.7	9354.6	121.1	0.937

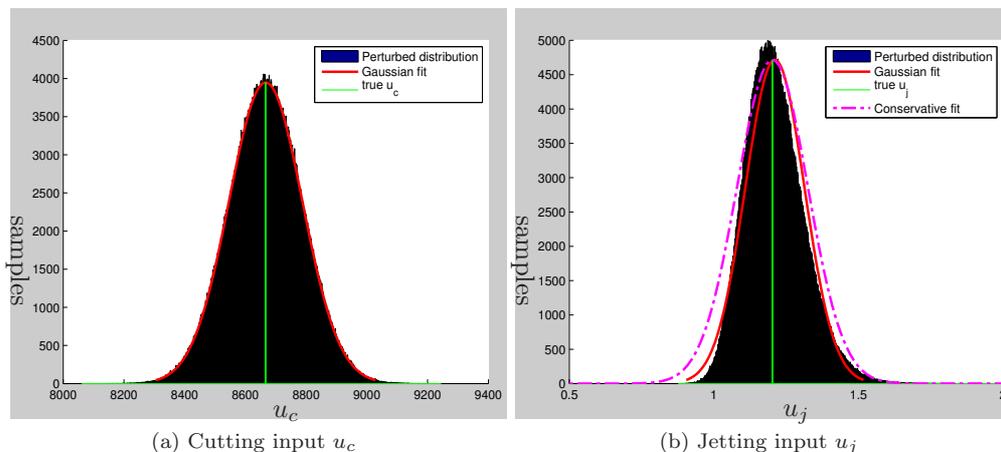


Figure A.1: Scenario A. Monte Carlo experiment showing the distribution of  $10^6$  random samples of  $u_c$  corrupted with noise (left) and  $u_c$  corrupted with noise (right). The red curve shows the Gaussian fit to the sample, the green line indicates the true values of  $u_c$  and  $u_j$ , respectively. The purple curve (right) shows the conservative Gaussian fit.

equal to  $\alpha = 0.05$ . The results of the tests are reported in Table A.1. First of all, notice the high p-values [Sellke et al., 2001] of the Kolmogorov-Smirnov tests. Indeed, in all of the cases the p-values are much higher than the rejection threshold  $p_{rej} = 0.05$ . These, combined with the large number of samples in the Monte Carlo experiment, indicate a good fit of the Gaussian to the empirical distribution. Next, we see that the difference between the true value of cutting input  $u_c$  and the mean of Monte Carlo samples is below 0.01% of the nominal value of  $u_c$ . Thus, we can safely assume that the uncertainty in the cutting input  $u_c$  is zero-mean. Finally,

Table A.2: Results of Monte Carlo experiments for different values of the jetting input  $u_j$ .

true value of $u_j$	sample mean	sample std	fitted std	conservative std
1.2026	1.2111	0.1021	0.1021	0.1225
1.2454	1.2548	0.1097	0.1097	0.1317
1.4623	1.4776	0.1539	0.1539	0.1846
1.5191	1.5366	0.1672	0.1672	0.2006
1.8670	1.9008	0.2612	0.2612	0.3134
1.9603	1.9995	0.2917	0.2917	0.3501
1.9609	2.0007	0.2920	0.2920	0.3504
1.9706	2.0112	0.2952	0.2952	0.3542
2.0459	2.0916	0.3211	0.3211	0.3853
2.0949	2.1438	0.3384	0.3384	0.4061
2.1442	2.1973	0.3569	0.3569	0.4283
2.4305	2.5109	0.4805	0.4805	0.5766
2.8580	2.9975	0.7429	0.7178	0.8614
2.9245	3.0744	0.8413	0.7597	0.9116
3.0647	3.2424	1.3157	0.8588	1.0305
3.1665	3.3612	7.7742	0.9603	1.1524

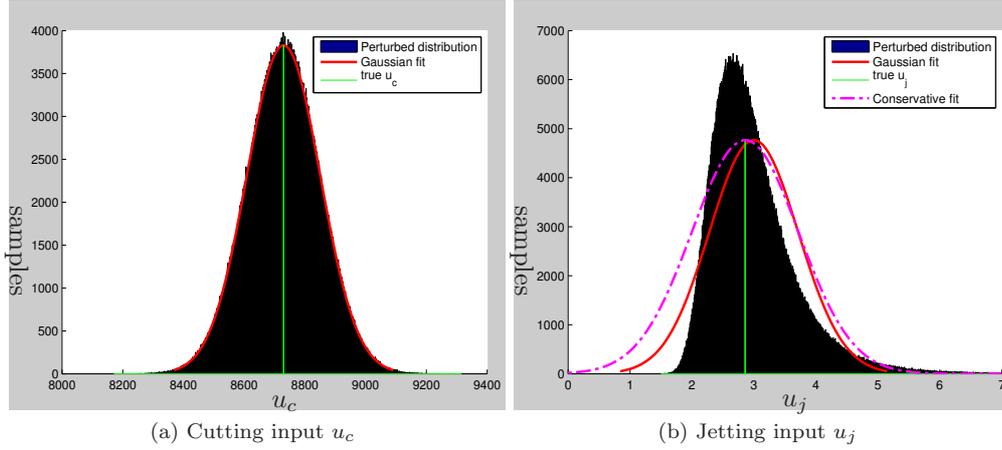


Figure A.2: Scenario B. Monte Carlo experiment showing the distribution of  $10^6$  random samples of  $u_c$  corrupted with noise (left) and  $u_c$  corrupted with noise (right). The red curve shows the Gaussian fit to the sample, the green line indicates the true values of  $u_c$  and  $u_j$ , respectively. The purple curve (right) shows the conservative Gaussian fit.

the standard deviations of the fitted Gaussian are almost indifferent (relative to the magnitude of  $u_c$ ) to the true value of  $u_c$ . Therefore, it is reasonable to approximate all the sampled std reported in the Table A.1 by the average of the sampled std, i.e., by  $\sigma = 120.3$ .

To conclude, the results of the Monte Carlo experiment described above give a strong support to the claim that the noise in the cutting input  $u_c$  can be approximated by the zero-mean Gaussian with the standard deviation given by  $\sigma = 120.3$ .

What can be further observed from Figures A.1b–A.3b is that the uncertainty in the jetting input  $u_j$  has non-Gaussian characteristics. This is further confirmed by Kolmogorov-Smirnov test of normality that rejects the normality hypothesis with p-value always lower than  $10^{-8}$ .

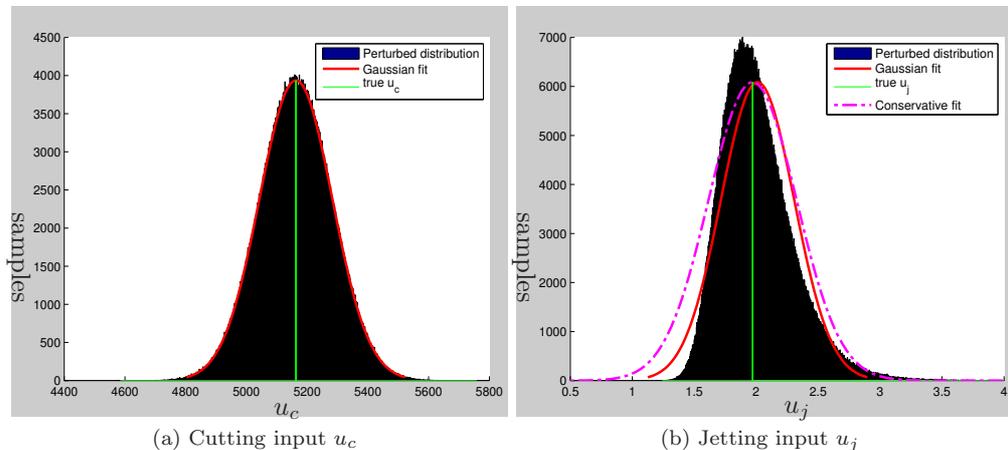


Figure A.3: Scenario C. Monte Carlo experiment showing the distribution of  $10^6$  random samples of  $u_c$  corrupted with noise (left) and  $u_c$  corrupted with noise (right). The red curve shows the Gaussian fit to the sample, the green line indicates the true values of  $u_c$  and  $u_j$ , respectively. The purple curve (right) shows the conservative Gaussian fit.

The samples obtained from the Monte Carlo experiment form a shape that resembles a Gamma distribution, however the gamma fits are also rejected by the Kolmogorov-Smirnov test, although with larger p-values than in the case of normality hypothesis. Therefore, we conclude that an accurate parametric approximation of the noise in  $u_j$  leads to a complicated statistical problem that is beyond the scope of this thesis. Instead, we decided to model the noise in  $u_j$  by a zero-mean Gaussian with the standard deviation that is large enough to cover the “thick tails” of the empirical distribution that can be seen in Figures A.1b–A.3b. Such an operation results in a conservative estimate of the uncertainty in the jetting input  $u_j$  since it increases the std of the “true” noise. However, this rise in the std is not very significant as can be seen in Figures A.1b–A.3b. Furthermore, the Gaussian approximation of the uncertainty distribution of  $u_j$  is very useful in the estimation algorithms. Table A.2 gives an overview of the  $u_j$ -related outcomes of the Monte Carlo experiment. It reports the true values of the jetting input  $u_j$  with sampled mean and sampled std, fitted std, which is obtained from the data with outliers filtered out, and the conservative std, which is equal to 120% of the fitted std.

As can be observed in the case of jetting input  $u_j$  the corresponding noise depends on the true value of  $u_j$ . This is different from the case of cutting input  $u_c$  where we established that the noise has constant variance. However, given the Table A.2, we can establish an empirical relation between the values of the jetting input  $u_j$  and the corresponding standard deviations. To the empirical data we fitted a polynomial of type

$$y = a + b \cdot x^c, \quad (\text{A.2})$$

where  $a$ ,  $b$ , and  $c$  are free parameters to be determined by the optimization algorithm. The choice of such a structure of the fitted curve is motivated by the model of the jetting input  $u_j$  (2.16b). We can observe in Figure A.4 that such a polynomial, with appropriate parameters  $a$ ,  $b$ , and  $c$ , accurately fits the empirical data.

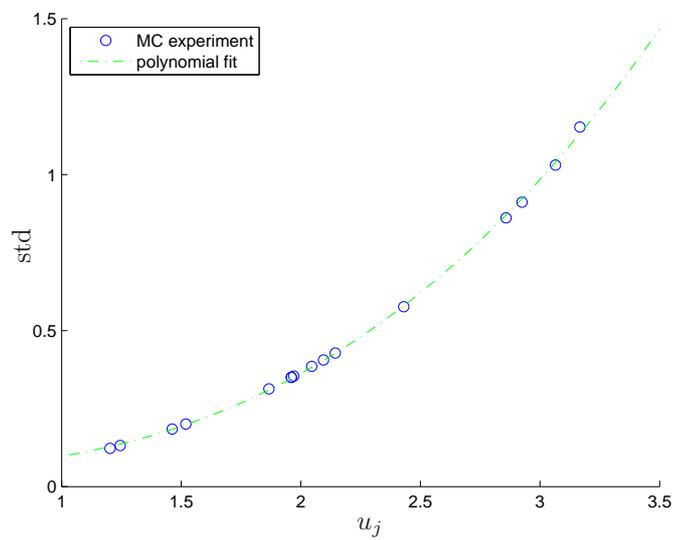


Figure A.4: The relation between the true value of jetting input  $u_j$  and the standard deviation of the Gaussian fitted to data obtained from the Monte Carlo experiment. The green curve is a polynomial (A.2) fitted to the empirical data. The parameters of the polynomial are:  $a = 0.04874$ ,  $b = 0.04857$ ,  $c = 2.693$ .



## Appendix B

# Proof of Proposition 7.1

In this appendix we prove Proposition 7.1 that states the following:

**Proposition B.1.** *If  $X$  is a normally distributed random variable with mean  $\mu$  and variance  $\sigma^2$  and  $C_1, C_2$  are certain positive constants then the probability density function of the variable*

$$C_1 \max\left(0, 1 - \frac{C_2}{X^2}\right) \quad (\text{B.1})$$

is given by

$$p_{C_1 \max(0, 1 - \frac{C_2}{X^2})}(x) = \sqrt{\frac{C_1 C_2}{2\pi\sigma^2(C_1 - x)^3}} e^{-\frac{C_1 C_2 + \mu^2}{2\sigma^2}} \cosh\left(\frac{\mu\sqrt{C_1 C_2}}{\sigma^2\sqrt{C_1 - x}}\right) \mathbf{1}_{(0, C_1]}(x) \quad (\text{B.2a})$$

$$+ \left(1 - \int_0^1 \sqrt{\frac{C_2}{2\pi\sigma^2 y^3}} e^{-\frac{C_2 + \mu^2}{2\sigma^2}} \cosh\left(\frac{\mu\sqrt{C_2}}{\sigma^2\sqrt{y}}\right) dy\right) \delta_0(x), \quad (\text{B.2b})$$

where  $\mathbf{1}_{(0, C_1]}$  is an indicator function and  $\delta_0$  is the Dirac delta.

In order to prove Proposition B.1 it is convenient to start with deriving the following two technical lemmas:

**Lemma B.1.** *If  $X$  is a normally distributed random variable with mean  $\mu$  and variance  $\sigma^2$  then the probability density function of the variable  $X^2$  is given by*

$$p_{X^2}(x) = \frac{1}{\sqrt{2\pi\sigma^2 x}} e^{-\frac{x+\mu^2}{2\sigma^2}} \cosh\left(\frac{\mu\sqrt{x}}{\sigma^2}\right). \quad (\text{B.3})$$

*Proof.* Note that the cumulative distribution function of the variable  $X^2$  is given by

$$\mathbb{P}(X^2 \leq x) = \mathbb{P}(X \leq \sqrt{x}) - \mathbb{P}(X \leq -\sqrt{x}) \quad (\text{B.4a})$$

$$= \Phi\left(\frac{\sqrt{x} - \mu}{\sigma}\right) - \Phi\left(\frac{-\sqrt{x} - \mu}{\sigma}\right), \quad (\text{B.4b})$$

where  $\Phi$  is the cumulative distribution function of the standard normal variable. Since  $\Phi$  is smooth at every point the probability density function of  $X^2$  can be found by taking the first derivative of the right hand side of the equation (B.4b)

$$p_{X^2}(x) = \frac{d}{dx} \left( \Phi\left(\frac{\sqrt{x} - \mu}{\sigma}\right) - \Phi\left(\frac{-\sqrt{x} - \mu}{\sigma}\right) \right) \quad (\text{B.5a})$$

$$= \frac{1}{2\sqrt{2\pi\sigma^2x}} \left( e^{-\frac{(\sqrt{x}-\mu)^2}{2\sigma^2}} + e^{-\frac{(\sqrt{x}+\mu)^2}{2\sigma^2}} \right) \quad (\text{B.5b})$$

$$= \frac{1}{\sqrt{2\pi\sigma^2x}} e^{-\frac{x+\mu^2}{2\sigma^2}} \frac{1}{2} \left( e^{\frac{\mu\sqrt{x}}{\sigma^2}} + e^{-\frac{\mu\sqrt{x}}{\sigma^2}} \right) \quad (\text{B.5c})$$

$$= \frac{1}{\sqrt{2\pi\sigma^2x}} e^{-\frac{x+\mu^2}{2\sigma^2}} \cosh\left(\frac{\mu\sqrt{x}}{\sigma^2}\right). \quad (\text{B.5d})$$

□

**Lemma B.2.** *If  $X$  is a normally distributed random variable with mean  $\mu$  and variance  $\sigma^2$  and  $C$  is a positive constant then the probability density function of the variable  $\frac{C}{X^2}$  is given by*

$$p_{\frac{C}{X^2}}(x) = \sqrt{\frac{C}{2\pi\sigma^2x^3}} e^{-\frac{\frac{C}{x} + \mu^2}{2\sigma^2}} \cosh\left(\frac{\mu\sqrt{C}}{\sigma^2\sqrt{x}}\right). \quad (\text{B.6})$$

*Proof.* The variable  $\frac{C}{X^2}$  takes only positive values, hence we can restrict the domain of the probability density function to the interval  $(0, +\infty)$ . Let  $x > 0$ . Then

$$\mathbb{P}\left(\frac{C}{X^2} \leq x\right) = 1 - \mathbb{P}\left(X^2 \leq \frac{C}{x}\right), \quad (\text{B.7})$$

and, since the cumulative distribution of  $X^2$  is smooth, we find the probability density function

of the variable  $\frac{C}{X^2}$  by differentiating both sides of the equation (B.7):

$$p_{\frac{C}{X^2}}(x) = \frac{C}{x^2} p_{X^2}\left(\frac{C}{x}\right). \quad (\text{B.8})$$

Finally, the formula (B.6) is obtained by applying Lemma B.1 to the last equation.  $\square$

*Proof of Proposition B.1.* The variable  $C_1 \max\left(0, 1 - \frac{C_2}{X^2}\right)$  introduced in (B.1) takes values in the interval  $[0, C_1]$ . It is continuously distributed on the interval  $(0, C_1]$  and has a singularity at zero, where a positive probability mass is concentrated. Therefore, for  $x > 0$  the probability density function of  $C_1 \max\left(0, 1 - \frac{C_2}{X^2}\right)$  is equal to the probability density function of a variable  $C_1 - \frac{C_1 C_2}{X^2}$  and the probability mass at  $x = 0$  is equal to  $\mathbb{P}\left(1 - \frac{C_2}{X^2} \leq 0\right)$ . Let us now consider  $x > 0$ . The variable  $C_1 - \frac{C_1 C_2}{X^2}$  is continuous, hence again its density can be derived from the cumulative density function

$$p_{C_1 \max\left(0, 1 - \frac{C_2}{X^2}\right)}(x) = \frac{d}{dx} \mathbb{P}\left(C_1 - \frac{C_1 C_2}{X^2} \leq x\right) \quad (\text{B.9a})$$

$$= \frac{d}{dx} \left(1 - \mathbb{P}\left(\frac{C_1 C_2}{X^2} \leq C_1 - x\right)\right) \quad (\text{B.9b})$$

$$= p_{\frac{C_1 C_2}{X^2}}(C_1 - x). \quad (\text{B.9c})$$

For  $x = 0$  we have

$$\mathbb{P}\left(1 - \frac{C_2}{X^2} \leq 0\right) = 1 - \mathbb{P}\left(\frac{C_2}{X^2} \leq 1\right) \quad (\text{B.10a})$$

$$= 1 - \int_{-\infty}^1 p_{\frac{C_2}{X^2}}(y) dy. \quad (\text{B.10b})$$

Note that the integral in (B.10b) is in fact the integral from zero to one, since  $\frac{C_2}{X^2}$  is a positive random variable. Therefore, the density of  $C_1 \max\left(0, 1 - \frac{C_2}{X^2}\right)$  is given by

$$p_{C_1 \max\left(0, 1 - \frac{C_2}{X^2}\right)}(x) = p_{\frac{C_1 C_2}{X^2}}(C_1 - x) \mathbf{1}_{(0, C_1]}(x) \quad (\text{B.11a})$$

$$+ \left(1 - \int_0^1 p_{\frac{C_2}{X^2}}(y) dy\right) \delta_0(x). \quad (\text{B.11b})$$

Finally, applying Lemma B.2 to the last equation leads to the desired formula (B.2b).  $\square$



# Nomenclature

## Roman Symbols

$a_k$	Update operator at time step $k$
$b_k$	Prediction operator at time step $k$
$\bar{c}^N$	Improved resampling operator
$c^N$	Multinomial sampling operator
$h_c$	Cutting depth
$k_{ch}$	Cutting force coefficient (horizontal)
$k_{cv}$	Cutting force coefficient (vertical)
$F_{vc}$	Visor cylinder force
$W_d$	Width of the drag-head
$h_z$	Dredging depth
$e_{ch}$	Process noise associated to the variable $k_{ch}$
$e_d$	Process noise associated to the variable $d_m$
$e_m$	Process noise associated to the variable $m_s$
$e_{F_{vc}}^o$	Measurement noise of the variable $F_{vc}$
$e_s^o$	Measurement noise of the variable $h_s$
$e_t^o$	Measurement noise of the variable $h_t$
$e_{vsh}^o$	Measurement noise of the variable $v_{sh}$
$k_e$	Erosion coefficient
$e_s$	Process noise associated to the variable $h_s$
$e_{vh}$	Process noise associated to the variable $k_{vk}$
$e_{vsh}$	Process noise associated to the variable $v_{sh}$

$e_{vc}$	Process noise associated to the variable $F_{vc}$
$h_{ex}$	Total excavation depth
$\mathbf{F}_k$	Linear process model at time step $k$
$\mathbf{f}_k$	Process model at time step $k$
$F_k$	Unsaturated state model at time step $k$
$Q_{s,c}$	Sand flow loosened by the teeth
$Q_{w,j}$	Water flow produced by jets
$Q_{s,j}$	Sand flow loosened by the jets
$Q_m$	Production mixture flow
$Q_{w,t}$	Toe water flow from surroundings
$Q_{w,v}$	Water flow through the valve
$k_d$	Friction coefficient total drag force
$d_m$	Average grain diameter
$\mathbf{H}_k$	Linear observation model at time step $k$
$\mathbf{h}_k$	Observation model at time step $k$
$F_{ch}$	Horizontal cutting force
$\mathbf{I}$	Identity matrix
$\mathbf{H}$	Matrix used in Central-Difference approximation
$I_m$	$m$ -th order quadrature rule
$\mathbf{J}_k$	Information matrix at time step $k$
$h_j$	Jetting depth
$Q_{w,j}$	Jet water flow
$\mathbf{K}_k$	Kalman gain at time step $k$
$\tilde{K}_k$	Probability kernel used by the Saturated Particle Filter
$g_k$	Likelihood function at time step $k$
$\mathcal{U}$	Input space
$\mathcal{X}$	State space
$\mathcal{Y}$	Observation space
$r_{vc}$	Moment arm
$p_j$	Jet nozzle pressure

$N_j$	Number of jet nozzles
$Q_o$	Outgoing flow rate
$\mathbf{P}_{k k}$	Covariance of the Gaussian posterior $p(\mathbf{x}_k   \mathcal{Y}_k)$ at time step $k$
$p_0$	Initial distribution
$k_{si}$	In situ permeability
$k_k^N$	Particle filter at time step $k$
$\mathbf{Q}_k$	Covariance matrix of the state noise $\mathbf{v}_k$
$q_i$	Predicted probability of saturation of the $i$ -th particle
$q_i^\alpha$	Updated probability of saturation of the $i$ -th particle
$\mathbf{R}_k$	Covariance matrix of the observation noise $\mathbf{w}_k$
$k_{vh}$	Ratio between horizontal and vertical cutting force
$k_{vh}$	Ration between $k_{ch}$ and $k_{cv}$
$\mathbf{x}_k^i$	Representative point
$Re_p$	Reynolds number
$h_s$	Sand bed height
$m_s$	Mass of the sand bed
$f_e$	Scouring function
$f_s$	Settling function
$v_{sh}$	Speed of the ship
$F_{th}$	Thrust force of the propeller blades
$Q_i$	Incoming flow rate
$m_t$	Total mass of the mixture in the hopper
$V_t$	Total volume of the mixture in the hopper
$T_s$	Sampling time
$u_k$	Deterministic input at time step $k$
$\mathbf{v}_k$	System noise at time step $k$
$F_{cv}$	Vertical cutting force
$v_{s0}$	Undisturbed settling velocity of a single particle
$x_2$	Visor length
$h_v$	Visor depth

$\mathbf{w}_k$	Measurement noise at time step $k$
$\mathbf{x}_k$	State variable at time step $k$
$\hat{\mathbf{x}}_k$	Estimate of the state at time step $k$
$\mathbf{x}_0$	Initial state of the system
$\mathbf{y}_k$	Observation variable at time step $k$
$\mathcal{Y}_k$	Measurements up to time step $k$

### Greek Symbols

$\alpha^i$	Weight of the $i$ -th term in the Gaussian term
$q_i^\alpha$	Updated probability of saturation
$\alpha_{lt}$	Angle of the lower suction pipe
$\partial A$	Boundary of the set $A$
$\beta$	Richardson-Zakai exponent
$q_i$	Predicted probability of saturation
$\rho_i$	Density of the incoming mixture
$\kappa$	Parameter of the UKF
$\rho_m$	Density of the mixture in the hopper
$\mu^i$	Mean of the $i$ -th term in the Gaussian term
$\omega_k^i$	Weight corresponding to the representative point $\mathbf{x}_k^i$
$\omega_k(\cdot)$	Likelihood function at time step $k$
$\rho_o$	Density of the outgoing mixture
$\rho_q$	Density of quartz
$\rho_i^{dh}$	Density of the mixture in the drag-head model
$\rho_s$	Sand bed density in the hopper
$\Sigma^i$	Covariance of the $i$ -th term in the Gaussian term
$\sigma_{ch}$	Standard deviation of the variable $e_{ch}$
$\sigma_d$	Standard deviation of the variable $e_d$
$\sigma_{vc}$	Standard deviation of the variable $e_{vc}$
$\sigma_m$	Standard deviation of the variable $e_m$
$\sigma_{Fvc}^o$	Standard deviation of the variable $e_{Fvc}^o$
$\sigma_s^o$	Standard deviation of the variable $e_s^o$

$\sigma_t^o$	Standard deviation of the variable $e_t^o$
$\sigma_{vsh}^o$	Standard deviation of the variable $e_{vsh}^o$
$\sigma_s$	Standard deviation of the variable $e_s$
$\sigma_{vh}$	Standard deviation of the variable $e_{vh}$
$\sigma_{vsh}$	Standard deviation of the variable $e_{vsh}$
$\Sigma_{\mathcal{X}}$	Collection of measurable subsets of the state space $\mathcal{X}$
$\tau_t$	Transport delay in the mixture density $\rho_i^{dh}$
$\alpha_v$	Visor angle
$\rho_w$	Density of water

### Other Symbols

$\nabla_x$	First-order derivative operator with respect to the variable $x$
$\Delta_x^y$	Second-order derivative operator with respect to the variables $x$ and $y$
$\mathbf{D}_k^{11}$	Matrix used for computation of PCRB
$\mathbf{D}_k^{12}$	Matrix used for computation of PCRB
$\mathbf{D}_k^{22}$	Matrix used for computation of PCRB
$\mathbb{E}_{\mathbf{x}}$	Expectation operator taken with respect to the variable $\mathbf{x}$
$\mathbb{E}$	Expectation operator
$\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$	Probability density function of a Gaussian random variable with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ evaluated at $x$
$p(\mathbf{x}_k   \mathcal{Y}_k)$	Conditional probability density of the state $\mathbf{x}_k$ given the measurements $\mathcal{Y}_k$

### Acronyms

BF	Bayesian Filter
BPF	Bootstrap Particle Filter
CBPF	Constrained Bootstrap Particle Filter
CDF	Central Difference Filter
CRB	Cramér-Rao Bound
CSPF	Convex Saturated Particle Filter
CSSDS	Convex Saturated Stochastic Dynamical System
EKF	Extended Kalman Filter
FPF	Feedback Particle Filter
GHF	Gauss-Hermite Filter

GSF Gaussian Sum Filter  
IEKF Iterated Extended Kalman Filter  
iSPF Improved Saturated Particle Filter  
KF Kalman Filter  
LRKF Linear Regression Kalman Filter  
MAP Maximum A Posteriori estimator  
MMSE Minimum Mean-Square Error estimator  
MPC Model Predictive Controller  
MSE Mean-Square Error estimator  
PCRB Posterior Cramér-Rao Bound  
PDF Probability Density Function  
PF Particle Filter  
RMSE Root Mean Squared Error  
ROPF Reduced-Order Particle Filter  
SDS Stochastic Dynamical System  
SPF Saturated Particle Filter  
SSDS Saturated Stochastic Dynamical System  
TSHD Trailing Suction Hopper Dredger  
UBPF Unconstrained Bootstrap Particle Filter  
UKF Unscented Kalman Filter  
UT Unscented Transformation

## Summary

A Trailing Suction Hopper Dredger (TSHD) is a ship that excavates sediments from the sea bottom while sailing. In situ material is excavated with a special tool called the Drag-Head, then it is hydraulically transported through a pipe to the hopper where it is temporarily stored. After the dredging is completed the collected material is transported and discharged at a specified location. The efficiency of this process is highly dependent on the detailed knowledge of the excavated soil.

The optimization of dredging operations is of vital importance for future improvement in efficiency, accuracy and from the viewpoint of labor saving. The automated onboard systems that have been developed to optimize the dredging performance require knowledge of several uncertain soil-dependent parameters. These cannot be directly measured but have to be estimated online from the available measurements. Such estimation is a challenging task due to lack of sufficient sensors, severe nonlinearities in models, and time-varying nature of the parameters of interest.

In this thesis we focus on two of the most important TSHD-related models. These are:

- I. Drag-Head Model - describing the excavation process,
- II. Hopper Model - describing the sedimentation process occurring inside the hopper.

They contain several uncertain soil-dependent parameters that need to be estimated. These are:

- I. horizontal cutting force coefficient  $k_{ch}$  (*Drag-Head Model*),
- II. ratio  $k_{vh}$  between the horizontal and vertical cutting forces (*Drag-Head Model*),
- III. in situ permeability  $k_{si}$  (*Drag-Head Model*),
- IV. average grain diameter  $d_m$  (*Hopper Model*).

Both processes, together with the corresponding estimation problems, are discussed in detail in Chapter 2.

The highly uncertain and time-varying nature of the soil-dependent parameters and the nonlinear dynamics of the models used to describe dredging process make the estimation a challenging task. The algorithms that are capable of tackling these type of problems are Nonlinear Bayesian Filters (NBF). In Chapter 3 we review several types of NBF, namely:

- I. parametric filters based on the Taylor series expansion (EKF, IEKF),
- II. parametric filters based on statistical approximations (UKF, GHF, CDF),
- III. parametric filters based on Gaussian Sum approximations (GSF),
- IV. nonparametric filters based on the importance sampling (BPF),

## V. nonparametric filters based on the mean-field control-oriented approach (FPF).

In Chapter 4 we investigate the applicability of these nonlinear filters to the estimation problems that originate from the Drag-Head Model. The problems are: the Cutting Estimation Problem and the Cutting and Jetting Estimation Problem. The Cutting Estimation Problem applies for any cutting excavation tool whereas the Cutting and Jetting Estimation Problem is applicable only for tools equipped with cutting and jetting components. The former problem considers estimation of the ratio  $k_{vh}$  between cutting forces and the horizontal cutting force coefficient  $k_{ch}$ , the latter problem deals with the estimation of the horizontal cutting force coefficient  $k_{ch}$  and the in situ permeability  $k_{si}$ . To solve the aforementioned estimation problems one needs to handle time-varying delay in the measurement of incoming density  $\rho_i$ , which is discussed separately.

It is concluded that among the tested methods the best solution to the Cutting Estimation Problem is provided by the CDF and, in case of large uncertainty in the initial states, by the GSF. To solve the Cutting and Jetting Estimation Problem it is crucial to exploit the correlation between the horizontal cutting force coefficient  $k_{ch}$  and the in situ permeability  $k_{si}$ . This is done by a cascaded filter, which uses the PF to obtain an estimate of  $k_{si}$ , which will be further filtered by a Steady State Identification (SSI) filter, and finally by the BF to produce a final estimate of  $k_{ch}$ .

In Chapter 5 we develop a novel class of nonlinear particle filters: the Saturated Particle Filter (SPF) that is used to solve the Hopper Estimation Problem. The SPF is a general method designed for Saturated Stochastic Dynamical Systems (SSDS), which are severely nonlinear systems often used in modeling real-life problems. They are characterized by a constrained probability distribution exhibiting singularity on the boundary of the saturation region. Such singularities make it difficult to estimate the states or the parameters of SSDSs by standard nonlinear filters. Our new method exploits the specific structure of the SSDS in order to design an importance sampling distribution that accounts for the most recent measurements in the prediction step of the filtering algorithm.

Chapter 6 deals with the asymptotic properties of the SPF. We establish the conditions under which the SPF converges to the optimal theoretical filter. The convergence of our method is closely related to the appropriate resampling scheme. This led to the development of the improved Saturated Particle Filter (iSPF) which combines the importance sampling of the SPF with a novel resampling algorithm.

In Chapter 7 the iSPF together with other nonparametric methods from Chapter 3 are used to estimate the average grain diameter  $d_m$ , which solves the Hopper Estimation Problem. Because the sedimentation process is naturally divided into three regimes, to find the most efficient filtering method we considered each mode separately. We conclude that:

- I. for the No-Overflow loading phase the best estimate of  $d_m$  is obtained by the FPF,
- II. for the Overflow loading phases with weak erosion, the recommended filtering method is the Reduced-Order PF,
- III. for the Overflow loading phases with strong erosion, the best estimation performance is achieved by the Reduced-Order PF when the excavated soil is fine and the Hybrid SPF when the excavated soil is coarse.

The final solution to the Hopper Estimation Problem is obtained by integrating the filters designed for separate modes into a global estimator.

Chapter 8 concludes the thesis.

## Samenvatting

Een sleephopperzuiger is een schip dat sediment van de zeebodem opgraaft tijdens het varen. Het sediment wordt opgegraven met behulp van een speciale Draghead (sleepkop). Het opgegraven sediment wordt vervolgens met behulp van een hydraulische pomp door een buis naar de hopper (laadruim) getransporteerd waar het tijdelijk wordt opgeslagen. Nadat het baggerproces klaar is, wordt het opgeslagen sediment op een specifieke locatie gelost. De efficiëntie van dit baggerproces hangt sterk af van de kennis van het opgegraven sediment. Het optimaliseren van de baggerprocessen is van vitaal belang om in de toekomst de efficiëntie te verhogen, de nauwkeurigheid te verbeteren en arbeid te besparen. Geautomatiseerde systemen aan boord die speciaal zijn ontwikkeld om het baggeren te optimaliseren, vereisen kennis van verschillende onzekere bodemafhankelijke parameters. De onzekere parameters kunnen niet rechtstreeks worden gemeten, maar moeten uit beschikbare metingen worden geschat. Een uitdaging bij het schatten van deze parameters is een gebrek aan voldoende sensoren, sterk niet-lineaire modellen en het tijdsafhankelijke karakter van de parameters.

Dit proefschrift richt zich op twee van de belangrijkste sleephopperzuiger-gerelateerde modellen, namelijk:

- I. Draghead Model - beschrijft het opgravingsproces,
- II. Hopper Model - beschrijft het sedimentatieproces in de hopper.

Deze modellen bevatten een aantal onzekere bodemafhankelijke parameters die moeten worden geschat. Deze parameters zijn:

- I. de horizontale snijkrachtcoëfficiënt  $k_{ch}$  (*Draghead Model*),
- II. de ratio  $k_{vh}$  tussen de horizontale en verticale snijkrachten (*Draghead Model*),
- III. de in situ permeabiliteit  $k_{si}$  (*Draghead Model*),
- IV. de gemiddelde korreldiameter  $d_m$  (*Hopper Model*).

Beide modellen, samen met de bijbehorende schattingsproblematiek, worden in detail besproken in hoofdstuk 2.

Het zeer onzekere en tijdsafhankelijke karakter van de bodemafhankelijke parameters en de niet-lineaire dynamica van de modellen die worden gebruikt om het baggerproces te beschrijven zorgen ervoor dat de schatting een zeer uitdagende opgave is. De algoritmen die in staat zijn deze uitdagingen op te lossen zijn niet-lineaire Bayesiaanse filters (NBF). In hoofdstuk 3 worden een aantal NBF variaties beschreven, namelijk:

- I. parametrische filters op basis van de Taylor-serie expansie (EKF, IEKF),
- II. parametrische filters op basis van statistische benaderingen (UKF, GHF, CDF),

- III. parametrische filters op basis van Gauss Sum benaderingen (GSF),
- IV. niet-parametrische filters op basis van importance sampling (BPF),
- V. niet-parametrische filters op basis van de mean-field control-oriented approach (FPF).

In hoofdstuk 4 wordt de toepasbaarheid van de niet-lineaire filters op de schattingsproblemen die voortkomen uit het Draghead Model onderzocht. De problemen zijn het Cutting Estimation Problem en het Cutting and Jetting Estimation Problem. Het Cutting Estimation Problem geldt voor elke snijknop, terwijl het Cutting and Jetting Estimation Problem alleen van toepassing is op werktuigen uitgerust met snij- en jetting componenten. Het eerstgenoemde probleem beschouwt schattingen van de verhouding  $k_{vh}$  tussen snijkrachten en de horizontale snijkrachtcoëfficiënt  $k_{ch}$ , terwijl het tweede probleem de schattingen van de horizontale snijkrachtcoëfficiënt  $k_{ch}$  en in situ permeabiliteit  $k_{si}$  beschouwt. Om de hiervoor genoemde schattingsproblemen op te lossen moet men omgaan met een tijdsvariërende vertraging bij de meting van de inkomende materiaaldichtheid  $\rho_i$ . Dit laatste wordt afzonderlijk besproken.

Er wordt geconcludeerd dat onder de geteste methoden de beste oplossing voor het Cutting Estimation Problem wordt verkregen door gebruik te maken van de CDF. In het geval van grote onzekerheid in de begintoestand wordt het beste resultaat verkregen door de GSF. Om het Cutting and Jetting Estimation Problem op te lossen is het cruciaal om de correlatie tussen de horizontale snijkrachtcoëfficiënt  $k_{ch}$  en de in situ permeabiliteit  $k_{si}$  te benutten. Dit wordt gedaan door een cascade filter dat de PF gebruikt om een schatting te maken van  $k_{si}$ . Vervolgens wordt dit resultaat met een Steady State Identification (SSI) filter verder gefilterd en tenslotte kan door het gebruikmaken van de BF een uiteindelijke schatting van  $k_{ch}$  worden verkregen.

In hoofdstuk 5 wordt een nieuwe categorie van niet-lineaire particle filters ontwikkeld: de Saturated Particle Filter (SPF) dat wordt gebruikt om het Hopper Estimation Problem op te lossen. De SPF is een algemene methode ontwikkeld voor Saturated Stochastic Dynamical Systems (SSDS) dat vaak wordt toegepast om sterk praktische en niet-lineaire systemen te modelleren. Ze worden gekenmerkt door een beperkte kansverdeling die een singulariteit op de grens van het verzadigingsbereik laten zien. Dergelijke singulariteiten maken het moeilijk om de status of de parameters van SSDS te schatten met behulp van standaard niet-lineaire filters. De nieuwe methode maakt gebruik van de specifieke structuur van de SSDS om een importance sampling distribution te ontwerpen die in de predictiestap van het filteralgoritme rekening houdt met de meest recente metingen.

Hoofdstuk 6 gaat over de asymptotische eigenschappen van de SPF. De voorwaarden waaronder de SPF convergeert naar een optimale theoretische filter worden vastgesteld. De convergentie van deze methode is nauw verwant aan het juiste resampling schema. Dit leidde tot de ontwikkeling van de improved Saturated Particle Filter (iSPF) waarin importance sampling van de SPF met een nieuw resampling algoritme wordt gecombineerd.

In hoofdstuk 7 worden iSPF samen met andere niet-parametrische methoden uit hoofdstuk 3 gebruikt om de gemiddelde korreldiameter  $d_m$  te schatten, zodat het Hopper Estimation Problem wordt opgelost. Omdat het sedimentatieproces in het

laadruim van nature verdeeld zijn in drie regimes, is om de meest efficiënte filtermethode te vinden elke toestand afzonderlijk beschouwd. De volgende conclusies zijn hieruit getrokken:

- I. voor de laadfase zonder overvloeiverliezen wordt de beste schatting van  $d_m$  verkregen door de FPF,
- II. voor de laadfase met overvloeiverliezen en zwakke erosie wordt de Reduced-Order PF filtermethode aangeraden,
- III. voor de laadfase met overvloeiverliezen, sterke erosie en fijn sediment wordt de beste schatting gegeven door de Reduced-Order PF en voor grof sediment door de Hybrid SPF.

De uiteindelijke oplossing voor het Hopper Estimation Problem wordt verkregen door het integreren van de ontworpen filters voor de afzonderlijke regimes tot een globaal schattingsalgoritme.

Hoofdstuk 8 sluit dit proefschrift af.

## Curriculum Vitae

Pawel Mirosław Stano was born on March 4<sup>th</sup> 1983 in Kraków, Poland. He graduated from the technical school of electrical engineering in 2003, and, in the same year was accepted for MSc program in mathematics at Jagiellonian University in Kraków. For the whole duration of the program he was receiving a scholarship for outstanding grades. In 2007, he worked at HTA Consulting as an analyst responsible for decision modeling and cost-effectiveness analysis of cancer treatments. In August 2007, he moved to Amsterdam after being accepted for the joint *Short Track Master* program between his Alma Mater and Vrije Universiteit Amsterdam, and awarded the VU Fellowship for outstanding students. In 2008, he started working at Vrije Universiteit Amsterdam as a research assistant of prof. dr. Aad van der Vaart within the project *Priors for Markov chains and hidden Markov models*. In the same year he received the M.Sc. degree (cum laude) in applied mathematics from Jagiellonian University and M.Sc. degree (cum laude) in stochastic mathematics from Vrije Universiteit Amsterdam. For his master thesis *Statistical tests for Markov chains and Hidden Markov Models* the Polish Mathematical Society awarded him with the second prize in the national contest for the *Best Student's Work on the Probability Theory and Applications of Mathematics*. In January 2009, he joined Delft Center for Systems and Control at Delft University of Technology to conduct his PhD research under supervision of prof. dr. Robert Babuška. His project was in a close collaboration with the IHC Systems B.V., Sliedrecht and aimed to solve nonlinear estimation problems for hopper dredgers. Within his PhD project, he spent the first months of 2012 as a visiting scholar at the Coordinated Science Laboratory, University of Illinois, Urbana-Champaign in the group of prof. Prashant Mehta. Since February 2013 he is employed at IHC Systems B.V., where he continues his research on online estimators used onboard modern hopper dredgers. In his free time he enjoys scuba diving, backpacking and writing.



# References

- <http://www.mathworks.com/matlabcentral/fileexchange/35468-particle-filter-tutorial>. URL <http://www.mathworks.com/matlabcentral/fileexchange/35468-particle-filter-tutorial>. 67
- B. M. Akesson, J. B. Jorgensen, N. K. Poulsen, and S. B. Jorgensen. A Generalized Autocovariance Least-Squares Method for Kalman Filter Tuning. *Journal of Process Control*, 18: 769–779, 2008. 36
- S. Ali-Löytty. Efficient Gaussian Mixture Filter for Hybrid Positioning. In *IEEE/ION Position Location and Navigation Symposium*, pages 60–66, Monterey, California, 2008a. 40
- S. Ali-Löytty. On the Convergence of the Gaussian Mixture Filter. Technical Report 89, Tampere University of Technology, Department of Mathematics, 2008b. 55, 57
- D. Alspach. Comments on “On the Identification of Variances and Adaptive Kalman Filtering”. *IEEE Transactions on Automatic Control*, 17:843–845, 1972. 36
- D. Alspach and H. Sorenson. Recursive Bayesian Estimation Using Gaussian Sums. *Automatica*, 7:465–479, 1971. 53, 54
- D. Alspach and H. Sorenson. Nonlinear Bayesian Estimation Using Gaussian Sum Approximations. *IEEE Transactions on Automatic Control*, 17:439–448, 1972. 53
- B. D. O. Anderson and J. B. Moore. *Optimal Filtering*. NJ: Prentice-Hall, 1979. 32, 36
- E. H. Aoki, A. Bagchi, P. Mandal, and Y. Boers. A Theoretical Analysis of Bayes-optimal Multi-target Tracking and Labelling. Technical report, University of Twente, the Netherlands, 2011. URL <http://www.math.utwente.nl/publications>. 32
- I. Arasaratnam and S. Haykin. Cubature Kalman Filters. *IEEE Transactions on Automatic Control*, 54:1254–1269, 2009. 30, 46
- I. Arasaratnam, S. Haykin, and R. Elliott. Discrete-Time Nonlinear Filtering Algorithms Using Gauss-Hermite Quadrature. *Proceedings of the IEEE*, 95:953–977, 2007. 47, 53, 54, 55, 57
- I. Arasaratnam, S. Haykin, and T. R. Hurd. Cubature Kalman Filtering for Continuous-Discrete Systems: Theory and Simulations. *IEEE Transactions on Signal Processing*, 58(10): 4977–4993, 2010. 46
- M. S. Arulampalam, B. Ristic, N. Gordon, and T. Mansell. Bearings-Only Tracking of Manoeuvring Targets Using Particle Filters. *EURASIP Journal on Applied Signal Processing*, 15:2351–2365, 2004. 112

- 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*, 50: 174–188, 2002. [30](#), [32](#), [61](#), [96](#), [101](#), [108](#), [136](#)
- R. Atar. Exponential Decay Rate of the Filter’s Dependence on the Initial Distribution. In *The Oxford Handbook of Nonlinear Filtering*. Oxford University Press, 2011. [35](#)
- Y. Bar-Shalom, X.-Rong Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. Wiley-Interscience, 2001. [31](#)
- R. Barbieri, L. M. Frank, D. P. Nguyen, M. C. Quirk, V. Solo, M. A. Wilson, and E. N. Brown. Dynamic Analyses of Information Encoding in Neural Ensembles. *Neural Computation*, 16: 277–307, 2004. [29](#)
- B. M. Bell and F. W. Cathey. The Iterated Kalman Filter Update as a Gauss-Newton Method. *IEEE Transactions on Automatic Control*, 38:294–297, 1993. [40](#)
- V. E. Beneš. Exact Finite-Dimensional Filters for Certain Diffusions with Nonlinear Drift. *Stochastics*, 5:65–92, 1981. [30](#), [32](#)
- T. R. Bewley and A. S. Sharma. Efficient Grid-Based Bayesian Estimation of Nonlinear Low-Dimensional Systems with Sparse Non-Gaussian PDFs. *Automatica*, 48:1286–1290, 2012. [30](#), [32](#)
- S. Bhat and D. Saraf. Steady State Identification, Gross Error Detection, and Data Reconciliation for Industrial Process Units. *Industrial and Engineering Chemistry Research*, 43: 4323–4336, 2004. [87](#)
- I. Bilik and J. Tabrikian. Target Tracking in Glint Noise Environment Using Nonlinear Non-Gaussian Kalman Filter. In *Proceedings of IEEE International Radar Conference*, pages 282–286, Verona, New York, 2006. [52](#)
- H. A.P. Blom and E. A. Bloem. Decomposed Particle Filtering and Track Swap Estimation in Tracking Two Closely Spaced Targets. In *Proceedings of the 14th International Conference on Information Fusion (FUSION)*, pages 1–8, Chicago, Illinois, 2011a. [32](#)
- H. A.P. Blom and E. A. Bloem. Optimal Decomposed Particle Filtering of Two Closely Spaced Gaussian Targets. In *Proceedings of the IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, pages 7895–7901, Orlando, Florida, 2011b. [30](#)
- S. Bolognani, L. Tubiana, and M. Zigliotto. Extended Kalman Filter Tuning in Sensorless PMSM Drives. *IEEE Transactions on Industry Applications*, 39:1741–1747, 2003. [40](#)
- J. Braaksma. *Model-Based Control of Hopper Dredgers*. PhD thesis, Delft University of Technology, 2008. [1](#), [8](#), [11](#), [16](#), [20](#), [21](#), [25](#), [143](#), [147](#), [148](#), [150](#), [152](#)
- J. Braaksma, J. B. Klaassens, R. Babuška, and C. de Keizer. Model Predictive Control for Optimizing the Overall Dredging Performance of a Trailing Suction Hopper Dredger. In *Proceedings of the Eighteenth World Dredging Congress (WODCON XVIII)*, pages 1263–1274, 2007a. [1](#), [8](#)
- J. Braaksma, J. B. Klaassens, R. Babuška, and C. de Keizer. A Computationally Efficient Model for Predicting Overflow Mixture Density in a Hopper Dredger. *Terra et Aqua*, 106: 16–25, 2007b. [1](#), [8](#), [16](#), [18](#), [19](#), [22](#), [23](#), [151](#)

- J. Braaksma, J. Osnabrugge, R. Babuška, C. de Keizer, and J. B. Klaassens. Artificial Intelligence on Board of Dredgers for Optimal Land Reclamation. In *Proceedings of the CEDA Dredging Days*, pages 1–14, Rotterdam, 2007c. [1](#), [8](#)
- A. Budhiraja. Feller and Stability Properties of the Nonlinear Filter. In *The Oxford Handbook of Nonlinear Filtering*. Oxford University Press, 2011. [35](#)
- A. Budhiraja, L. Chen, and C. Lee. A Survey of Numerical Methods for Nonlinear Filtering Problems. *Physica D*, 230:27–36, 2007. [64](#)
- G. Burgers, P. J. van Leeuwen, and G. Evensen. Analysis Scheme in the Ensemble Kalman Filter. *Monthly Weather Review*, 126:1719–1724, 1998. [30](#)
- T. R. Camp. Sedimentation and the Design of Settling Tanks. *Transactions of the ASCE*, pages 895–936, 1946. [16](#), [20](#)
- O. Cappé and E. Moulines. On-line Expectation-Maximization Algorithm for Latent Data Models. *Journal of the Royal Statistical Society Series B*, 71:593–613, 2009. [31](#)
- O. Cappé, E. Moulines, and T. Rydén. *Inference in Hidden Markov Models*. Springer, 2009. [31](#)
- B. P. Carlin, N. G. Polson, and D. S. Stoffer. A Monte Carlo Approach to Nonnormal and Nonlinear State-Space Models. *Journal of the American Statistical Association*, 87:493–500, 1992. [96](#)
- J. Carpenter, P. Clifford, and P. Fearnhead. Improved Particle Filter for Nonlinear Problems. *IEEE-Proceedings Radar, Sonar and Navigation*, 146:2–7, 1999. [63](#)
- C. Chang and J. Tabaczynski. Application of State Estimation to Target Tracking. *IEEE Transactions on Automatic Control*, 29:98–109, 1984. [40](#)
- P. Chigansky, R. Lipster, and R. Van Handel. Intrinsic Methods in Filter Stability. In *The Oxford Handbook of Nonlinear Filtering*. Oxford University Press, 2011. [35](#)
- H. Cramer. *Mathematical Methods of Statistics*. Princeton University Press, Princeton, New Jersey, 1946. [33](#)
- D. Crisan and B. Rozovskii. *The Oxford Handbook of Nonlinear Filtering*. Oxford University Press, 2011. [35](#)
- D. Cristian and A. Doucet. A Survey of Convergence Results on Particle Filtering Methods for Practitioners. *IEEE Transactions on Signal Processing*, 50:736–746, 2002. [30](#), [62](#), [64](#), [96](#), [121](#), [122](#), [124](#), [126](#)
- J. Danielsson. Stochastic Volatility in Asset Prices Estimation with Simulated Maximum Likelihood. *Journal of Econometrics*, 64:375–400, 1994. [29](#)
- G. Dantzig. *Linear Programming and Extensions*. Princeton University Press, 1998. [105](#)
- F. E. Daum. Exact Finite-Dimensional Nonlinear Filters. *IEEE Transactions on Automatic Control*, 31:616–622, 1986. [30](#), [33](#)
- F. E. Daum. Nonlinear Filters: Beyond the Kalman Filter. *IEEE A&E Systems Magazine*, 20(8):57–69, 2005. [30](#), [33](#)

- A. Doucet, S. Godsill, and C. Andrieu. On Sequential Monte Carlo Sampling Methods for Bayesian Filtering. *Statistics and Computing*, 10:197–208, 2000. [30](#), [61](#), [62](#)
- A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag New York, 2001. [64](#), [121](#), [122](#)
- J. Duník, M. Šimandl, and O. Straka. Adaptive Choice of Scaling Parameter in Derivative-Free Local Filters. In *Proceedings of the 13th International Conference on Information Fusion (FUSION)*, pages 1–8, Edinburgh, 2010. [46](#)
- G. Evensen. The Ensemble Kalman Filter: Theoretical Formulation and Practical Implementation. *Ocean Dynamics*, 53:343–367, 2003. [30](#), [66](#)
- G. Evensen. *Data Assimilation: The Ensemble Kalman Filter*. Springer-Verlag, Berlin, 2006. [30](#), [66](#)
- A. Farina, B. Ristic, and D. Benvenuti. Tracking a Ballistic Target: Comparison of Several Nonlinear Filters. *IEEE Transactions on Aerospace and Electronic Systems*, 38:854–867, 2002. [33](#), [34](#), [57](#), [58](#)
- R. Di Felice. The Sedimentation Velocity of Dilute Suspensions of Nearly Mono-Sized Spheres. *International Journal of Multiphase Flow*, 25:559–574, 1999. [16](#)
- W. Feller. On the Kolmogorov-Smirnov Limit Theorems for Empirical Distributions. *Annals of Mathematical Statistics*, 19:177–189, 1948. [173](#)
- G. B. Folland. *Real analysis: modern techniques and their applications*. Pure and applied mathematics. Wiley, 1984. [107](#)
- X. Fu and Y. Jia. An Improvement on Resampling Algorithms of Particle Filters. *IEEE Transactions on Signal Processing*, 58:5414–5420, 2010. [62](#), [63](#)
- J.-L. Gauvain and C.-H. Lee. Maximum a Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chains. *IEEE Transactions on Speech and Audio Processing*, 2:291–298, 1994. [32](#)
- W. R. Gilks and C. Berzuini. Following a Moving Target - Monte Carlo Inference for Dynamic Bayesian Models. *Journal of the Royal Statistical Society*, 63:127–146, 2001. [96](#), [112](#)
- F. Le Gland, F. Monbet, and V. D. Tran. Large Sample Asymptotics for the Ensemble Kalman Filter. In D. Crisan and B. Rozovskii, editors, *The Oxford Handbook of Nonlinear Filtering*, chapter 22, pages 598–631. Oxford University Press, 2011. [30](#), [35](#)
- S. J. Godsill, A. Doucet, and M. West. Monte Carlo Smoothing for Nonlinear Time Series. *Journal of the American Statistical Association*, 99(465):156–168, 2004. [31](#)
- G. H. Golub and J. H. Welsch. Calculation of Gauss Quadrature Rules. *Mathematics of Computation*, 23:221–230, 1969. [48](#)
- N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation. *IEE-Proceedings-F*, 140:107–113, 1993. [62](#), [112](#)
- S. J. Gosh, D. Roy, and C. S. Manohar. New Forms of Extended Kalman Filter via Transversal Linearization and Applications to Structural System Identification. *Computer Methods in Applied Mechanics and Engineering*, 196:5063–5083, 2007. [41](#)

- U. Hahlbrock and I. Freese. Hopper Load Measurement; Some Recent Experiences with a Remote-Operated Data Acquisition and Analysing System. *Terra et Aqua*, 72:18–26, 1998. 1
- S. Haykin. *Adaptive Filter Theory*. Prentice-Hall, 1991. 29, 35
- M. Hernandez, B. Ristic, A. Farina, and L. Timmoneri. A Comparison of Two Cramer-Rao Bounds for Nonlinear Filtering with  $p_d < 1$ . *IEEE Transactions on Signal Processing*, 52:2361–2370, 2004. 34
- Y. C. Ho and R. C. Lee. Bayesian Approach to Problems in Stochastic Estimation and Control. *IEEE Transactions on Automatic Control*, 9:333–339, 1964. 32, 35
- A. Honkela. Approximating Nonlinear Transformations of Probability Distributions for Nonlinear Independent Component Analysis. In *Proceedings of the International Joint Conference on Neural Networks*, volume 3, pages 2169–2174, Budapest, Hungary, 2004. 47
- J. T. Horwood and A. B. Poore. Adaptive Gaussian Sum Filters for Space Surveillance. *IEEE Transactions on Automatic Control*, 56:1777–1790, 2011. 55, 57
- M. Huang, P. E. Caines, and R. P. Malhame. Large-population Cost-coupled LQG Problems with Nonuniform Agents: Individual-mass Behavior and Decentralized  $\epsilon$ -Nash Equilibria. *IEEE Transaction on Automatic Control*, 52:1560–1571, 2007. 64
- S. Van Huffel and J. Vandewalle. *The Total Least Squares Problem: Computational Aspects and Analysis*. Society for Industrial and Applied Mathematics, 1991. 93
- M. Hurzeler and H. Kunsch. Monte Carlo Approximations for General State Space Models. *Journal of Computational and Graphical Statistics*, 7:175–193, 1998. 63
- <http://nft.kky.zcu.cz/tools>. Identification and Decision Making Research Group at the University of West Bohemia. URL <http://nft.kky.zcu.cz/tools>. 67
- T. Ikeda, H. Kato, T. Tomoi, S. Yano, and Y. Nakamura. Artificial Intelligence for Dredging Control of the Trailing Suction Hopper Dredger. In *Proceedings of the 14th World Dredging Congress (WODCON XIV)*, 1995. 1, 8
- E. L. Ionides, C. Breto, and A. A. King. Inference for Nonlinear Dynamical Systems. *Proceedings of the National Academy of Sciences*, 103:18438–18443, 2006. 14, 23
- K. Ito and K. Xiong. Gaussian Filters for Nonlinear Filtering Problems. *IEEE Transactions on Automatic Control*, 45:910–927, 2000. 30, 43, 45, 49, 55, 57
- A. Jazwinski. *Stochastic Processing and Filtering Theory*. Academic Press, 1970. 40
- S. Julier. The Scaled Unscented Transform. In *Proceedings of the American Control Conference*, volume 6, pages 4555–4559, Anchorage, Alaska, 2002. 46, 47
- S. Julier and J. Uhlmann. A General Method for Approximating Nonlinear Transformations of Probability Distributions. Technical report, Robotics Research Group, Department of Engineering Science, University of Oxford, 1996. 40, 43, 46, 47, 66
- S. Julier and J. Uhlmann. Reduced Sigma Point Filters for the Propagation of Means and Covariances Through Nonlinear Transformations. In *Proceedings of the American Control Conference*, volume 2, pages 887–892, Anchorage, Alaska, 2002. 47

- S. Julier and J. Uhlmann. Unscented Filtering and Nonlinear Estimation. *Proceedings of the IEEE*, 92:401–422, 2004a. [30](#), [46](#), [47](#)
- S. Julier and J. Uhlmann. Corrections to Unscented Filtering and Nonlinear Estimation. *Proceedings of the IEEE*, 92:1958, 2004b. [30](#), [46](#), [47](#)
- S. Julier, J. Uhlmann, and H. F. Durrant-Whyte. A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators. *IEEE Transactions on Automatic Control*, 45:477–482, 2000. [43](#), [44](#), [46](#)
- R. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME - Journal of Basic Engineering*, 82:35–45, 1960. [32](#), [35](#)
- T. Kerr. Status of CR-Like Lower Bounds for Nonlinear Filtering. *IEEE Transactions on Aerospace and Electronic Systems*, 25:590–600, 1989. [33](#)
- S. Kim, G. Eyink, J. Restrepo, F. Alexander, and G. Johnson. Ensemble Filtering for Nonlinear Dynamics. *Monthly Weather Review*, 131:2586–2594, 2003. [66](#)
- G. Kitagawa. Non-Gaussian State-Space Modeling of Nonstationary Time Series. *Journal of the American Statistical Association*, 82:1032–1041, 1987. [31](#)
- G. Kitagawa. Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models. *Journal of Computational and Graphical Statistics*, 5:1–25, 1996. [62](#)
- G. Kitagawa. A Self-Organizing State-Space Model. *Journal of the American Statistical Association*, 93:1203–1215, 1998. [14](#), [23](#)
- M. L. Kleptsyna and A. Yu. Veretennikov. On Filtering with Unspecified Initial Data for Nonuniformly Ergodic Signals. In *The Oxford Handbook of Nonlinear Filtering*, pages 267–298. Oxford University Press, 2011. [33](#), [35](#), [62](#)
- A. Kong, J. S. Liu, and W. H. Wong. Sequential Imputations and Bayesian Missing Data Problems. *Journal of the American Statistical Association*, 89:278–288, 1994. [62](#)
- J. H. Kotecha and P. M. Djurić. Gaussian Particle Filters. *IEEE Transactions on Signal Processing*, 51:2592–2601, 2003a. [62](#)
- J. H. Kotecha and P. M. Djurić. Gaussian Sum Particle Filters. *IEEE Transactions on Signal Processing*, 51:2602–2612, 2003b. [62](#), [82](#)
- S. Kramer and H. Sorenson. Recursive Bayesian Estimation Using Piece-wise Constant Approximations. *Automatica*, 24:789–801, 1988. [30](#)
- S. G. Krantz and H. R. Parks. *Geometric Integration Theory*. Cornerstones Series. Birkhäuser Boston, 2008. [115](#)
- H. Künsch. Recursive Monte Carlo Filters: Algorithms and Theoretical Analysis. *The Annals of Statistics*, 33:1983–2021, 2005. [63](#), [121](#)
- I. Kurita, Y. Okayama, T. Tomoi, and Y. Nakamura. Automatic Operation System for Drag Suction Hopper Dredger. In *13th World Dredging Congress*, 1992. [1](#)
- H. J. Kushner and P. Dupuis. *Numerical Methods for Stochastic Control Problems In Continuous Time*. Applications of Mathematics. Springer-Verlag, New York, 2001. [33](#), [36](#), [38](#)

- I. Kyriakides, D. Morrel, and A. Papandreou-Suppappola. A Particle Filtering Approach to Constrained Motion Estimation in Tracking Multiple Targets. In *The 39th Asilomar Conference on Signals, Systems and Computers*, volume 28, pages 94–98, Pacific Grove, California, 2005. 96
- L. Lang, W. Chen, B. R. Bakshi, P. K. Goel, and S. Ungarala. Bayesian Estimation via Sequential Monte Carlo Sampling - Constrained Dynamic Systems. *Automatica*, 43:1615–1622, 2007. 96
- D. S. Lee and N. K. K. Chia. A Particle Algorithm for Sequential Bayesian Parameter Estimation and Model Selection. *IEEE Transactions on Signal Processing*, 50:326–336, 2002. 62, 63
- J. M. Lee. *Introduction to Topological Manifolds*. Graduate Texts in Mathematics. Springer, 2010. 106
- T. Lefebvre, H. Bruyninckx, and J. Schutter. Kalman Filters for Nonlinear Systems: A Comparison of Performance. *International Journal of Control*, 77:639–653, 2004. 41, 42, 43, 48, 51
- M. Lei, B. van Wyk, and Y. Qi. Online Estimation of the Approximate Posterior Cramer-Rao Lower Bound for Discrete-Time Nonlinear Filtering. *IEEE Transactions on Aerospace and Electronic Systems*, 47:37–57, 2011. 34
- Zs. Lendek, R. Babuška, J. Braaksma, and C. de Keizer. Decentralized Estimation of Overflow Losses in a Hopper Dredgers. *Control Engineering Practice*, 16:392–406, 2008. 18, 25, 143, 148, 159
- X. R. Li and V. P. Jilkov. Survey of Maneuvering Target Tracking, Part i: Dynamic Models. *IEEE Transactions on Aerospace and Electronic Systems*, 39:1333–1364, 2003. 57, 58
- D. V. Lindley. The Theory of Queues with a Single Server. *Mathematical Proceedings of the Cambridge Philosophical Society*, 48:277–289, 1952. 102
- J. S. Liu and R. Chen. Sequential Monte Carlo Methods for Dynamic Systems. *Journal of the American Statistical Association*, 93:1032–1044, 1998. 62
- L. Ljung and S. Gunnarsson. Adaptation and Tracking in System Identification - A Survey. *Automatica*, 26:7–21, 1990. 32, 35
- F. Massey. The Kolmogorov-Smirnov Test for Goodness of Fit. *Journal of the American Statistical Association*, 46:68–78, 1951. 173
- [www.mathworks.com/matlabcentral/fileexchange/18189](http://www.mathworks.com/matlabcentral/fileexchange/18189). Mathworks, a. URL <http://www.mathworks.com/matlabcentral/fileexchange/18189>. 67
- [www.mathworks.com/matlabcentral/fileexchange/18217](http://www.mathworks.com/matlabcentral/fileexchange/18217). Mathworks, b. URL <http://www.mathworks.com/matlabcentral/fileexchange/18217>. 67
- V. Matoušek. *Flow Mechanism of Sand-Water Mixtures in Pipelines*. PhD thesis, TU Delft, 1997. 20
- V. Matoušek. Hydraulic Transport as one of the Dredging Processes. Lecture Notes, 2001. 21, 22

- V. Matoušek. Pressure Drops and Flow Patterns in Sand-Mixture Pipes. *Experimental Thermal and Fluid Science*, 26:693–702, 2002. [1](#), [8](#)
- L. H. Matthies, R. Szeliski, and T. Kanade. Kalman Filter-based Algorithms for Estimating Depth from Image Sequences. *International Journal of Computer Vision*, 3:209–236, 1989. [32](#), [35](#)
- R. Mehra. On the Identification of Variances and Adaptive Kalman Filtering. *IEEE Transactions on Automatic Control*, 15:175–184, 1970. [35](#), [36](#)
- J. M. Mendel. *Lessons in Estimation Theory for Signal Processing, Communications, and Control*. Prentice Hall, 1995a. [35](#), [36](#)
- J. M. Mendel. *Lessons in Estimation Theory for Signal Processing, Communications, and Control*, chapter Iterated Least Squares and Extended Kalman Filtering. Prentice Hall, 1995b. [38](#)
- S. Meyn and R. Tweedie. *Markov Chains and Stochastic Stability*. Springer-Verlag London, 1993. [122](#)
- S. A. Miedema. The Cutting of Densely Compacted Sand Under Water. *Terra et Aqua*, 28:4–10, 1984. [1](#), [8](#)
- S. A. Miedema. *Calculation of the Cutting Forces when Cutting Water Saturated Sand*. PhD thesis, TU Delft, 1987. [11](#)
- S. A. Miedema. Modeling and Simulation of the Dynamic Behavior of a Pump/Pipeline Systems. In *Proceedings of the 17th Annual meeting and technical conference of the Western Dredging Association*, 1996. [11](#)
- S. Mirza and J. F. Richardson. Sedimentation of Suspensions of Particles of Two or More Sizes. *Chemical Engineering Science*, 34:447–454, 1979. [16](#)
- H. Morita, H. Nakamura, and M. Shimazaki. Control of Large Dredger with Oil Recovery Equipment. *IHI Engineering Review*, 35:53–58, 2002. [1](#)
- J. E. Mulquiney, J. P. Norton, A. J. Jakeman, and J. A. Taylor. Random Walks in the Kalman Filter: Implications for Greenhouse Gas Flux Deductions. *Environmetrics*, 6:473–478, 1995. [36](#)
- M. Murshed, B. Huang, and K. Nandakumar. Estimation and Control of Solid Oxide Fuel Cell System. *Computers and Chemical Engineering*, 34:96–111, 2010. [29](#)
- R. Niu, P. Willett, and Y. Bar-Shalom. Matrix CRLB Scaling due to Measurements of Uncertain Origin. *IEEE Transactions on Signal Processing*, 49:1325–1335, 2001. [34](#)
- M. Nørgaard, N. Poulsen, and O. Ravn. New Developments in State Estimation for Nonlinear Systems. *Automatica*, 36:1627–1638, 2000. [30](#), [48](#), [49](#)
- B. Øksendal. *Stochastic Differential Equations: An Introduction with Applications*. Springer Verlag, 2003a. [36](#), [65](#)
- B. Øksendal. *Stochastic Differential Equations: An Introduction with Applications*, chapter The Filtering Problem, pages 81–108. Springer Verlag, 2003b. [32](#), [64](#)

- S. C. Ooijens. Adding Dynamics to the Camp Model for the Calculation of Overflow Losses. *Terra et Aqua*, 76:12–21, 1999. [1](#), [8](#), [19](#)
- S. C. Ooijens, A. de Gruijter, A.A.H. Nieuwenhuijzen, and S. Vandycke. Research on Hopper Settlement Using Large-Scale Modelling. In *Proceedings of the CEDA Dredging Days*, pages 1–11, 2001. [16](#)
- M. Pitt and N. Shephard. Filtering via Simulation: Auxiliary Particle Filter. *Journal of the American Statistical Association*, 94:590–599, 1999. [63](#)
- K. Pomorski. Gauss-Hermite Approximation Formula. *Computer Physics Communications*, 174:181–186, 2006. [48](#)
- J. Prakash, S. C. Patwardhan, and S. L. Shah. Constrained Nonlinear State Estimation Using Ensemble Kalman Filters. *Industrial and Engineering Chemistry Research*, 49:2242–2253, 2010. [96](#), [105](#)
- W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in Fortran 77: The Art of Scientific Computing*. Cambridge University Press, New York, NY, USA, 2003. [115](#)
- V.S. Pugachev and I.N. Sinitsyn. *Stochastic Systems: Theory and Applications*. World Scientific, Singapore, 2001. [33](#), [38](#)
- M. Rajamani and J. Rawlings. Estimation of the Disturbance Structure from Data Using Semidefinite Programming and Optimal Weighting. *Automatica*, 45:142–148, 2009. [36](#)
- S. Rangan, A. Fletcher, and V. Goyal. Asymptotic Analysis of MAP Estimation via the Replica Method and Compressed Sensing. In *Proceedings of the 23rd Annual Conference on Neural Information Processing Systems*, pages 1545–1553, Vancouver, Canada, 2009. [32](#)
- C. R. Rao. Information and the Accuracy Attainable in the Estimation of Statistical Parameters. *Bulletin of Calcutta Mathematical Society*, 37:81–91, 1945. [33](#)
- H. E. Rauch, F. Tung, and C. T. Striebel. Maximum Likelihood Estimates of Linear Dynamic Systems. *AIAA Journal*, 3(8):1445–1450, 1965. [31](#)
- R. Reynolds. Robust Estimation of Covariance Matrices. *IEEE Transactions on Automatic Control*, 35:1047–1051, 1990. [36](#)
- J. Richardson and W. Zaki. Sedimentation and Fluidisation: Part i. *Transactions of the Institution of Chemical Engineers*, 32:35–53, 1954. [16](#), [19](#), [21](#), [23](#)
- B.D. Ripley. *Stochastic Simulation*. Wiley, New York, 1987. [42](#)
- B. Ristic, M. S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter: Particle Filters for Tracking Application*. Artech House, 2004a. [30](#), [32](#), [35](#), [38](#), [39](#), [40](#), [45](#), [62](#), [66](#), [96](#), [105](#)
- B. Ristic, M. S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter: Particle Filters for Tracking Application*, chapter Suboptimal Nonlinear Filters, pages 19–34. Artech House, 2004b. [38](#)
- J. Rodríguez-Millán and C. González. Three Mathematica Supported Proposals for the Discretization of Nonlinear Dynamical Control Systems. *Nonlinear Analysis*, 63:617–628, 2005. [30](#)

- S. Ross. *Probability and Statistics for Engineers and Scientists*. Elsevier Academic Press, 2009. 88
- P. N. Rowe. A Convenient Empirical Equation for Estimation of the Richardson-Zaki Exponent. *Chemical Engineering Science*, 43:2795–2796, 1987. 21, 22, 24
- S. Roweis and Z. Ghahramani. Learning Nonlinear Dynamical Systems using the Expectation-Maximization Algorithms. In Simon Haykin, editor, *Kalman Filtering and Neural Networks*, chapter 6, pages 175–220. Wiley, 2001. 31
- N. Saha and D. Roy. Extended Kalman Filters Using Explicit and Derivative-Free Local Linearizations. *Applied Mathematical Modelling*, 33:2545–2563, 2009. 41
- S. Sangsuk-Iam and T. Bullock. Analysis of Discrete-Time Kalman Filtering Under Incorrect Noise Covariances. *IEEE Transactions on Automatic Control*, 35:1304–1308, 1990. 36
- J. D. Sargan and A. Bhargava. Testing Residuals from Least Squares Regression for Being Generated by the Gaussian Random Walk. *Econometrica*, 51:153–174, 1983. 36
- S. Särkkä. Bayesian Estimation of Time-Varying Systems: Discrete-Time Systems. Written material for the course held in Spring 2012, 2012. URL [http://www.lce.hut.fi/~ssarkka/course\\_k2011/pdf/course\\_booklet\\_2011.pdf](http://www.lce.hut.fi/~ssarkka/course_k2011/pdf/course_booklet_2011.pdf). 30, 31
- S. Särkkä and J. Hartikainen. On Gaussian Optimal Smoothing of Non-Linear State Space Models. *IEEE Transaction on Automatic Control*, 55(8):1938–1941, 2010. 31
- T. S. Schei. A Finite-Difference Method for Linearization in Nonlinear Estimation Algorithms. *Automatica*, 33:2053–2058, 1997. 49
- T. Sellke, M. Bayarri, and J. Berger. Calibration of p-Values for Testing Precise Null Hypotheses. *The American Statistician*, 55:62–71, 2001. 174
- X. Shao, B. Huang, and J. M. Lee. Constrained Bayesian State Estimation - A Comparative Study and a New Particle Filter Based Approach. *Journal of Process and Control*, 20:143–157, 2010. 96, 100, 101, 108, 110, 118
- D. Simon. *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. Wiley-Interscience, 2006. 31
- F. Simonot. Convergence Rate for the Distributions of GI/M/1/n and M/GI/1/n as n Tends to Infinity. *Journal of Applied Probability*, 34:1049–1060, 1997. 102
- X. Song, P. Willett, and S. Zhou. Posterior Cramer-Rao Bounds for Doppler Biased Multistatic Range-only Tracking. In *Proceedings of the 14th International Conference on Information Fusion (FUSION)*, pages 1–8, Chicago, Illinois, 2011. 42
- K. Spingarn. Passive Position Location Estimation Using the Extended Kalman Filter. *IEEE Transactions on Aerospace and Electronic Systems*, 23:558–567, 1987. 41
- W. Stadje. A New Approach to the Lindley Recursion. *Statistics and Probability Letters*, 31:169–175, 1997. 96, 102
- P. Stano, Zs. Lendek, J. Braaksma, R. Babuška, and C. de Keizer. Particle Filters for Estimating Average Grain Diameter of Material Excavated by Hopper Dredger. In *Proceedings of IEEE Conference on Control Applications*, pages 292 – 297, Yokohama, Japan, 2010. 96, 127

- P. Stano, Zs. Lendek, and R. Babuška. Saturated Particle Filters. In *Proceedings of the American Control Conference*, pages 1819–1824, San Francisco, California, 2011. [96](#), [122](#)
- O. Straka, J. Dunik, and M. Simandl. Truncated Unscented Particle Filter. In *Proceedings of the American Control Conference*, pages 1825–1830, San Francisco, California, 2011. [96](#), [105](#)
- D. Talay. Simulation of Stochastic Differential Systems. *Lecture Notes in Physics: Probabilistic Methods in Applied Physics*, 451:54–96, 1995. [30](#)
- W. Ip Tam, K. N. Plataniotis, and D. Hatzinakos. An Adaptive Gaussian Sum Algorithm for Radar Tracking. *Signal Processing*, 77:85–104, 1999. [55](#), [57](#)
- H. Tanizaki. *Nonlinear Filters: Estimation and Applications*. Springer Berlin, 1996. [39](#), [40](#)
- H. Tanizaki and R. S. Mariano. Prediction Filtering and Smoothing in Non-Linear and Non-Normal Cases Using Monte Carlo Integration. *Journal of Applied Econometrics*, 9:163–179, 1994. [62](#)
- H. Tanizaki and R. S. Mariano. Nonlinear Filters based on Taylor Series Expansions. *Communications in Statistics: Theory and Methods*, 25:1261–1282, 1996. [40](#), [41](#)
- H. Tanizaki and R. S. Mariano. Nonlinear and Non-Gaussian State Space Modeling with Monte-Carlo Simulations. *Journal of Econometrics*, 83:263–290, 1998. [62](#)
- D. Tenne and T. Singh. The Higher Order Unscented Filter. In *Proceedings of the American Control Conference*, pages 2441–2446, Denver, Colorado, June 4-6 2003. [46](#)
- G. Terejanu, P. Singla, T. Singh, and P. D. Scott. A Novel Gaussian Sum Filter Method for Accurate Solution to the Nonlinear Filtering Problem. In *Proceedings of the 11th International Conference on Information Fusion*,, pages 1–8, Cologne, Germany, 2008. [55](#)
- P. Tichavsky, C. Muravchik, and A. Nehorai. Posterior Cramer-Rao Bounds for Discrete-Time Nonlinear Filtering. *IEEE Transactions on Signal Processing*, 46:1386–1396, 1998. [33](#), [34](#), [36](#), [37](#), [42](#)
- P. Vachhani, S. Narasimhan, and R. Rengaswamy. Robust and Reliable Estimation via Unscented Recursive Nonlinear Dynamic Data Reconciliation. *Journal of Process Control*, 16:1075–1086, 2006. [96](#), [105](#)
- R. van Handel. Nonlinear Filtering and Systems Theory. In *Proceedings of the 19th International Symposium on Mathematical Theory of Networks and Systems*, Budapest, Hungary, 2010. [35](#)
- C. van Rhee. Modelling the Sedimentation Process in a Trailing Suction Hopper Dredger. *Terra et Aqua*, 86:18–27, 2002a. [1](#), [8](#), [16](#)
- C. van Rhee. *On the sedimentation process in a Trailing Suction Hopper Dredger*. PhD thesis, TU Delft, 2002b. [1](#), [8](#), [16](#), [18](#), [19](#), [21](#)
- H. van Trees. *Detection, Estimation, and Modulation Theory: Part I*. Wiley, New York, 1968. [33](#)
- J. J. Verbeek, J. R. Nunnik, and N. Vlassis. Accelerated EM-based Clustering of Large Data Sets. *Data Mining Knowledge Discovery*, 13:291–307, 2006. [53](#)
- M. Verlaan and A. W. Heemink. Nonlinearity in Data Assimilation Applications: A Practical Method for Analysis. *Monthly Weather Review*, 129:1578–1589, 2001. [40](#)

- J. Vermaak, S. Maskell, and M. Briers. Online Sensor Registration. In *Proceedings of the IEEE Aerospace Conference*, pages 2117–2125, Big Sky, Montana, 2005. 54
- W. Vlasblom. Dredging Equipment and Technology. Lecture notes, 2003. URL <http://www.dredging.org/content.asp?page=105>. 12
- W. J. Vlasblom and S. A. Miedema. A Theory for Determining Sedimentation and Overflow Losses in Hoppers. In *Proceedings of the 14th World Dredging Congress*, pages 183–202, Amsterdam, 1995. 20, 22, 24
- M. Vlasiou, I. Adan, and J. Wessels. A Lindley-Type Equation Arising from a Carousel Problems. *Journal of Applied Probability*, 41:1171–1181, 2004. 102
- M. Šimandl and J. Duňík. Derivative-Free Estimation Methods: New Results and Performance Analysis. *Automatica*, 45:1749–1757, 2009. 31
- M. Šimandl, J. Královec, and P. Tichavsky. Filtering, Predictive, and Smoothing Cramér-Rao Bounds for Discrete-Time Nonlinear Dynamic Systems. *Automatica*, 37:1703–1716, 2001. 31, 34
- M. Šimandl, J. Královec, and T. Söderström. Anticipative Grid Design in Point-Mass Approach to Nonlinear State Estimation. *IEEE Transactions on Automatic Control*, 47:699–702, 2002. 30
- M. Šimandl, J. Královec, and T. Söderström. Advanced Point-Mass Method for Nonlinear State Estimation. *Automatica*, 42:1133–1145, 2006. 30
- E. A. Wan and R. Van Der Merwe. The Unscented Kalman Filter. In Simon Haykin, editor, *Kalman Filtering and Neural Networks*, chapter 7, pages 221–280. Wiley, 2001. 46
- E. A. Wan and R. van der Merwe. The Unscented Kalman Filter for Nonlinear Estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium*, pages 153–158, Lake Louise, Alberta, Canada, 2000. 43
- J. Weare. Particle Filtering with Path Sampling and an Application to a Bimodal Ocean Current Model. *Journal of Computational Physics*, 228:4312–4331, 2009. 29
- G. Welch and G. Bishop. [www.cs.unc.edu/~welch/kalman](http://www.cs.unc.edu/~welch/kalman). The University of Northern Carolina at Chapel Hill, Department of Computer Science. URL [www.cs.unc.edu/~welch/kalman](http://www.cs.unc.edu/~welch/kalman). 67
- G. Welch and G. Bishop. An Introduction to the Kalman Filter. Technical Report 95-041, Department of Computer Science, University of North Carolina at Chapel Hill., 1995. 32, 35, 36, 38
- J. L. Williams. Gaussian mixture reduction for tracking multiple maneuvering targets in clutter. Master’s thesis, Air Force Institute of Technology, Wright-Patterson Air Force Base, OH., 2003. Available: <http://sbg.mit.edu/jlwil/>. 57
- R.P. Wishner, J.A. Tabaczynski, and M. Athans. A Comparison of Three Non-Linear Filters. *Automatica*, 5:487–496, 1969. 41
- Y. Wu, M. Wu, D. Hu, and X. Hu. An Improvement to Unscented Transformation. In *Proceedings of the 17th Australian Joint Conference on Artificial Intelligence*, volume 3339, pages 1024–1029, Cairns, Australia, 2004. 47

- K. Xiong, H. Zhang, and L. Liu. Adaptive Robust Extended Kalman Filter for Nonlinear Stochastic Systems. *IET Control Theory Applications*, 3:239–250, 2008. [40](#)
- T. Yang, P. Mehta, and S. Meyn. A Mean-field Control-oriented Approach to Particle Filtering. In *Proceedings of the American Control Conference*, pages 2037–2043, San Francisco, California, 2011a. [30](#), [64](#), [65](#)
- T. Yang, P. G. Mehta, and S. P. Meyn. Feedback Particle Filter with Mean-field Coupling. In *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference.*, 2011b. [64](#), [65](#), [67](#)
- T. Yang, R. S. Laugesen, P. G. Mehta, and S. P. Meyn. Multivariable Feedback Particle Filter. In *IEEE Conference on Decision and Control*, 2012. [64](#), [65](#)
- H. Yin, P. G. Mehta, S. P. Meyn, and U. V. Shanbhag. Synchronization of Coupled Oscillators is a Game. In *Proceedings of the American Control Conference*, page 1783D1790, 2010. [64](#)
- P. Zarchan. *Tactical and Strategic Missile Guidance*, volume 199 of *Progress in astronautics and aeronautics*. American Institute of Aeronautics and Astronautics, 2002. [57](#)
- X. Zhang, P. Willett, and Y. Bar-Shalom. Dynamic Cramer-Rao Bound for Target Tracking in Clutter. *IEEE Transactions on Aerospace and Electronic Systems*, 41:1154–1167, 2005. [34](#)
- L. Zuo, R. Niu, and P. K. Varshney. Conditional Posterior Cramer-Rao Lower Bounds for Nonlinear Sequential Bayesian Estimation. *IEEE Transactions on Signal Processing*, 59: 1–14, 2011. [34](#)