

System Identification and Control of Rivers

Hasan Arshad Nasir

Submitted in total fulfilment of the requirements of the degree of
Doctor of Philosophy

National ICT Australia, NICTA
Department of Electrical and Electronic Engineering
THE UNIVERSITY OF MELBOURNE

September 2016

Copyright © 2016 Hasan Arshad Nasir

All rights reserved. No part of the publication may be reproduced in any form by print, photoprint, microfilm or any other means without written permission from the author.

Abstract

Water resource management has immense importance in the modern world. A large amount of water is wasted due to inefficient management of rivers, lakes and other water bodies associated with them. In order to achieve improved management of water resources, control and decision support systems can be employed

To design control systems for a river, a river model is required. Traditionally, the *Saint Venant equations* have been used for modelling purposes. The equations describe river flows accurately, however, they are complex, non-linear and require many unknown parameters. It is therefore difficult to use them for control design purposes. On the other hand, *data-based models* have proven to be very useful in control design for rivers. In this thesis, different data-based modelling methods are explored, and they are applied to the data from the upper part of Murray River in Australia. For each method, the thesis analyses the ease with which available prior knowledge can be incorporated in the modelling procedure and the ability of the obtained models to describe the river well.

For efficient river control, forecasts of future water demands and flows in the unregulated tributaries are required to be taken into account. A *Stochastic Model Predictive Control* (S-MPC) or a randomised version of it can not only accommodate such forecasts, but it can also handle physical and environmental constraints well. However, due to uncertainties in the forecasts, the feasibility of optimisation problems cannot always be guaranteed in the presence of constraints. This thesis proposes an S-MPC based river control schemes, that not only incorporate the forecasts, but also ensure feasibility of the optimisation problems. The schemes are successfully applied in simulations to the past data from the upper part of Murray River.

Another important aspect of river management is to mitigate flood risks. An ideal

strategy is to reduce the risk of severe floods, and at the same time not being overly cautious while performing normal river operations. This thesis uses *Value-at-Risk* (VaR) as a risk measure and incorporates it into the river control problem, forming a *Multiple Chance-Constrained optimisation Problem* (M-CCP), to be solved in an S-MPC setup. A computationally tractable *Optimisation and Testing algorithm* is developed to find solutions to M-CCPs, with probabilistic guarantees on the solution. The algorithm is successfully applied to the historical data from the upper part of Murray River. The simulation results show better regulation and flood avoidance.

Declaration

This is to certify that

1. the thesis comprises only my original work towards the PhD,
2. due acknowledgement has been made in the text to all other material used,
3. the thesis is less than 100,000 words in length, exclusive of tables, maps, bibliographies and appendices.

Hasan Arshad Nasir, September 2016

This page intentionally left blank.

Acknowledgements

First and foremost, special thanks to God Almighty for making me capable to write this PhD thesis. The journey has been much smoother than what I imagined in the beginning, and this is all because of His eternal blessings upon me.

It is difficult to fully acknowledge my PhD supervisor, Prof. Erik Weyer, in words. His support, encouragement and appreciation has gone a long way in this thesis, and for that, I owe him my deepest gratitude. His teachings will stay with me, and I will be always grateful to him for that. Apart from research, I also enjoyed our discussion on sports, especially cricket. In general, I would say it has been a great honour and pleasure working with him.

My thesis committee, which consists of Prof. Michael Cantoni and Dr. Su Ki Ooi, has been a very valuable resource during the course of my PhD. Their feedbacks and comments have always been very productive towards my work. I am extremely thankful to them for doing all of my evaluations and being a constructive reviewer of my work.

I would like to take this opportunity to acknowledge my collaborators: Dr. Algo Carè, and Associate Prof. Simone Garatti.

An year long collaborative work with Algo, in the development of randomised algorithm, has flourished a long way and contributed a lot to this thesis. I thank him for being always there to help me out with Mathematics and with any technical matters related to my PhD. It was a memorable experience working with Algo, especially because of his great sense of humour.

A two weeks long visit to Simone in Politechnic di Milano, in June 2014, opened a new passage for me to advance my thesis in the area of control. It was the time when I completed the system identification part of the thesis, and I was looking for a suitable

control strategy for rivers. He introduced the scenario based optimisation approach to me and enlightened me with its benefits. Later on, the approach became the foundation stone of my research work in the control of rivers.

A big thanks goes to National ICT Australia (NICTA) and The University of Melbourne for funding my postgraduate studies and providing me with the opportunity to attend conferences, workshops and seminars. I would also like to thank Murray Darling Basin Authority (MDBA), Canberra, who provided us the relevant Murray River data, needed for the research work carried out in this thesis. They have always been very gracious in sharing their experiences on river management. They also provided us an opportunity to visit the control facilities and weirs at Hume Reservoir and Lake Mulwala on Murray River, Australia. Also, I want to thank Carlton Connect Initiative Funds (CCIF), Australia, to partially sponsor the research and domestic travels.

I would also like to thank the administrative staffs in the Department of Electrical and Electronic Engineering and National ICT Australia (NICTA), for making my working experience enjoyable during my PhD. In particular, I thank Lyn Buchanan, Jackie Brissonette, David Benson, Nasrin Hashemi, Fang Chen, Natasha Baxter, Paula Thomas, Katie Lang and Ashleigh Nelson for their great support in handling my administrative tasks.

Also, a massive thanks goes to Whitley College in The University of Melbourne, who gave me the opportunity to be a residential tutor in the college and to let me practice my passion of teaching, during the second half of my PhD. Their support values a lot to me, and I would like to thank the residential leadership team of the college, which constitutes Margaret Welsford, Genevieve Leach and Jack Tan. Also, I want to thank Brenda Addie, who was the previous Academic Dean of the college.

I want to express my deepest gratitude to my parents, for whom, I do not possess the right words to acknowledge their support, as they have been the most supportive ones in my life. I will never forget the sacrifices that they have made on my education and the desires that they have for me to reach to the highest standards of knowledge. May God Almighty bless them always. My wife, Sabeen, has been such a great support to me since we have been together. I truly appreciate her quick adaptation with the life of a husband, who is a PhD student. I would also like to mention my elder brother, Ali, who

has always set great standards ahead of me in education. He has been very influential in my life since my childhood. Also, I cannot forget to mention my younger sister, Maryam, who is always willing to talk whenever I am stressed with work.

I thank all the teachers and supervisors that I have or had in my life. I can see their contributions to this thesis in different forms. A special mention goes to Associate Prof. Abubakr Muhammad, who was my supervisor in Lahore University of Management and Sciences, where I did my Masters degree in Computer Engineering. He helped me in enhancing my research abilities and prepared me to pursue a research degree, before joining The University of Melbourne.

Last but not least, I thank my friends and colleagues, inside and outside the university, for making a stimulating and friendly living environment for me in Melbourne. I truly appreciate their friendly discussions and enjoyable company.

This page intentionally left blank.

Preface

The thesis comprises materials and results which are obtained from the collaborative work of me and my supervisor, Prof. Erik Weyer, during the course of my postgraduate studies in The University of Melbourne, Australia. In addition to that, major portion of Chapter 4 is a joint work of me and my supervisor with Dr. Algo Carè, and some work in Chapter 3 was done in collaboration with Associate Prof. Simone Garatti.

All simulation results reported in the thesis are programmed in MATLAB[®]. For modelling purposes, CAPTAIN [1] and CONTSID [2] toolboxes are occasionally used in Chapter 2. The quadratic programming in the simulation of Stochastic Model Predictive Control (S-MPC) has been formulated using YALMIP [3] in MATLAB[®], and solved using SDPT3 [4]. The historical data of the upper part of Murray River, which is extensively used in this thesis, is provided by Murray Darling Basin Authority (MDBA) Australia.

The publications made during the course of my PhD are listed below.

Journal papers

1. **H.A. Nasir** and E. Weyer, "System identification of the Upper Part of Murray River", Control Engineering Practice (CEP-2016).

Conference papers

1. **H.A. Nasir**, A. Carè and E. Weyer, "A randomised approach to Multiple Chance-Constrained Problems (M-CCPs): An application to flood avoidance", Proceedings of the Conference on Decision and Control (CDC-2016).

2. **H.A. Nasir**, A. Carè and E. Weyer, "Control of rivers with flood avoidance", Proceedings of the Australian Control Conference (AuCC-2016), Newcastle, Australia.
3. **H.A. Nasir**, S. Garatti and E. Weyer, "Scenario based Stochastic MPC schemes for rivers with feasibility assurance", Proceedings of the European Control Conference (ECC-2016), Aalborg, Denmark.
4. **H.A. Nasir**, A. Carè and E. Weyer, "Randomised approach to flood control using Value-at-Risk", Proceedings of the Conference on Decision and Control (CDC-2015), Osaka, Japan.
5. **H.A. Nasir** and E. Weyer, "Estimation of models for the Upper Part of Murray River with flow dependent parameters", Proceedings of the IFAC Symposium on System Identification (SYSID-2015), Beijing, China.
6. **H.A. Nasir** and E. Weyer, "Control of a river stretch with uncertain inflows", Proceedings of the Australian Control Conference (AuCC-2014), Canberra, Australia.
7. **H.A. Nasir** and E. Weyer, "System identification of the Upper Part of Murray River", Proceedings of the European Control Conference (ECC-2014), Strasbourg, France.
8. **H.A. Nasir** and E. Weyer, "Comparison of Prediction Error Methods and Subspace Identification Methods for Rivers", Proceedings of the Australian Control Conference (AuCC-2013), Perth, Australia.

To my loving maa and paa

This page intentionally left blank.

Contents

1	Introduction	1
1.1	Importance of research on water management	1
1.2	Upper part of Murray River	3
1.3	Key steps to achieve better river management	8
1.3.1	Modelling	8
1.3.2	Control design	10
1.3.3	Flood avoidance	10
1.4	Literature review	11
1.4.1	Review on river modelling	11
1.4.2	Review on river control	14
1.4.3	Review on flood control	16
1.5	Thesis overview	18
1.6	Major contributions of the thesis	19
2	System Identification Methods Applied to the Upper Part of Murray River	21
2.1	Prelude	22
2.1.1	Operational objectives	23
2.1.2	Available prior information	24
2.1.3	Identification dataset	26
2.1.4	Time delay estimation	29
2.1.5	Estimating time delays: τ_H , τ_B and τ_P	30
2.1.6	Model orders and noise models	33
2.2	Modelling water level in Lake Mulwala	33
2.2.1	Prediction Error Method (PEM)	33
2.2.2	Maximum Likelihood (ML)	36
2.2.3	Continuous-Time (CT) identification	38
2.2.4	Data-Based Mechanistic (DBM) approach	40
2.2.5	Subspace Identification Method (SIM)	50
2.2.6	Comparison of results	55
2.3	Modelling water level in Lake Mulwala and flow at Doctors Point	57
2.3.1	PEM approach	57
2.3.2	ML approach	59
2.3.3	Continuous Time (CT) identification	64
2.3.4	SIM approach	67
2.4	Comparison of results	70

2.4.1	Simulation performance and suitability for control	71
2.4.2	Prior knowledge	74
2.4.3	Model type	75
2.4.4	Computational and software issues	75
Appendices		77
2.A	Additional figures	77
2.A.1	Flows of the validation data in Section 2.2	77
2.A.2	Simulation of the model in Eq. 2.6 with slight differences in τ_H	77
2.B	DBM based higher order model of water level in Lake Mulwala	80
2.C	Kiewa River modelling	83
2.D	Ovens River modelling	88
3	Scenario-based Stochastic Model Predictive Control (S-MPC) for Rivers	95
3.1	Model Predictive Control (MPC)	96
3.2	Stochastic MPC	98
3.2.1	State space representation for a finite time horizon	100
3.2.2	Feedback based control policy	101
3.2.3	Probabilistic constraints and Chance-Constrained optimisation Problem (CCP)	105
3.2.4	Scenario-based approach to CCPs	107
3.3	State space representation of rivers	108
3.3.1	State space model for a general river reach	108
3.3.2	Unregulated flow forecast	113
3.4	Stochastic MPC for rivers	115
3.4.1	Common constraints in rivers	115
3.4.2	Objective function	118
3.5	Stochastic MPC schemes for rivers with feasibility assurance	120
3.5.1	Scenario approach applied to the optimisation schemes	123
3.5.2	Feasibility of the scenario bases schemes	124
3.6	Application of the proposed schemes to the upper part of Murray River	125
3.6.1	Control design	127
3.6.2	Performance of Situation 1: (unknown unregulated inflows)	129
3.6.3	Performance of Situation 2: (unknown unregulated in- and out-flows)	133
3.6.4	Performance of Situation 3: (high unregulated inflows)	136
Appendices		143
3.A	Additional topics on control policies	143
3.A.1	A control policy which accommodates forecasts	143
3.A.2	Incorporating an integral action	144
3.B	Proof of Corollary 3.1	145
3.C	State space model of the water level in Lake Mulwala	146

4	A Randomised Approach to Multiple Chance-Constrained Problems	149
4.1	Preliminary concepts and problem statement	151
4.1.1	Loss function and loss distribution	151
4.1.2	Risk measures	152
4.1.3	Stochastic MPC based optimisation problem	156
4.2	Randomised approaches to solve an M-CCP with two chance-constraints .	159
4.2.1	Some possible ways to solve Problem (4.14)	159
4.2.2	Intuitive description of the <i>Optimisation and Testing algorithm</i>	161
4.3	Optimisation and Testing algorithm to solve an M-CCP with 2 chance-constraints	163
4.3.1	Auxiliary terms and procedures	163
4.3.2	Optimisation and Testing algorithm	172
4.4	M-CCPs with more than two chance-constraints - an extension	179
4.5	A simulation based example	186
4.5.1	Control objectives and control design	186
4.5.2	Simulation results	188
	Appendices	191
4.A	Proof of Theorem 4.2	191
5	Application of the Optimisation and Testing Algorithm to Flood Control	195
5.1	Flood risk measure and mitigation	196
5.1.1	Value-at-Risk as a flood risk measure	196
5.1.2	Stochastic MPC problem for flood avoidance	198
5.2	Application of the proposed algorithm to the upper part of Murray River	199
5.2.1	Control design	200
5.2.2	Performance of Situation 1: (unknown unregulated inflows)	202
5.2.3	Performance of Situation 2: (partially known future water demands)	208
6	Conclusions and Future Works	211
6.1	Conclusions	211
6.2	Future works	212

This page intentionally left blank.

List of Figures

1.1	Hume Reservoir to Lake Mulwala on the Murray River (plot not to scale).	4
1.2	Hume Reservoir gates.	4
1.3	Measuring station at Heywoods.	5
1.4	Yarrawonga Weir.	6
1.5	Top view of Lake Mulwala. Source: Google Maps, 2015.	6
2.1	Hume Reservoir to Lake Mulwala on the Murray River (plot not to scale).	23
2.2	Identification dataset.	28
2.3	Magnitude of the DFT of the water level measurements at Lake Mulwala from summer and winter datasets.	28
2.4	Cross-correlation coefficient, the maximum value gives the estimate of the time delay from Heywoods to Corowa (data is from 27 th Sep, 2002 to 3 rd Dec, 2002).	32
2.5	Cross-correlation coefficient, the maximum value gives the estimate of the time delay from Bandiana to Corowa (data is from 22 nd July, 2007 to 20 th Sep, 2007).	32
2.6	Simulation results on validation data with small variations in the water level in Lake Mulwala.	36
2.7	Simulation results on validation data with relatively large variations in the water level in Lake Mulwala.	37
2.8	Simulation results on validation data with small variations in the water level in Lake Mulwala.	40
2.9	Simulation results on validation data with relatively large variations in the water level in Lake Mulwala.	41
2.10	Non-parametric estimates of the state dependent parameters.	45
2.11	A non-parametric estimate of an SDP with straight-line fitting against a dataset from 2005.	47
2.12	A non-parametric estimate of an SDP with straight-line fitting against a dataset from 2011.	47
2.13	Simulation results on validation data with small variations in the water level in Lake Mulwala.	49
2.14	Simulation results on validation data with relatively large variations in the water level in Lake Mulwala.	50
2.15	Simulation results on validation data with small variations in the water level in Lake Mulwala.	54

2.16	Simulation results on validation data with relatively large variations in the water level in Lake Mulwala.	55
2.17	Simulation results on validation data with small variations in the water level in Lake Mulwala.	56
2.18	Simulation results on validation data with relatively large variations in the water level in Lake Mulwala.	57
2.19	Rating curve between the flow and water level at Doctors Point.	58
2.20	Identification dataset.	59
2.21	Simulation results on validation data.	60
2.22	Simulation results on validation data.	63
2.23	Simulation results on validation data.	63
2.24	Simulation results on validation data.	66
2.25	Simulation results on validation data.	67
2.26	Simulation results on validation data.	70
2.27	Simulation results on validation data.	71
2.28	Simulation results on validation data.	72
2.29	Simulation results on validation data.	72
2.30	Transfer function plots from each input flows to water level in Lake Mulwala.	74
2.31	Transfer function plots from each input flows to flow at Doctors Point. . .	75
2.32	Validation dataset from year 2003.	78
2.33	Validation dataset from year 2004.	78
2.34	Validation dataset from year 2002.	79
2.35	Validation dataset from year 2006.	79
2.36	Simulation of PEM based model.	80
2.37	Simulation results on validation data with small variations in the water level in Lake Mulwala.	82
2.38	Simulation results on validation data with relatively large variations in the water level in Lake Mulwala.	82
2.39	Kiewa River (plot not to scale).	83
2.40	Identification dataset.	85
2.41	Simulation results on a validation dataset.	85
2.42	Simulation results on a validation dataset.	86
2.43	Simulation results on a validation dataset.	87
2.44	Simulation results on a validation dataset.	87
2.45	Ovens River (plot not to scale).	89
2.46	Identification dataset.	90
2.47	Simulation results on a validation dataset.	91
2.48	Simulation results on a validation dataset.	91
2.49	Simulation results on a validation dataset.	92
2.50	Simulation results on a validation dataset.	93
3.1	Hume Reservoir to Lake Mulwala on the Murray River (plot not to scale).	126
3.2	Controlled water level using Scheme 1.	130
3.3	Regulated and unregulated flows using Scheme 1.	130
3.4	Zoomed outflows to Yarrawonga Main Channel and Mulwala Canal. . . .	130

3.5	Controlled water level using Scheme 2.	131
3.6	Regulated flows using Scheme 2.	132
3.7	Controlled water level using Scheme 2 with a non-zero \mathbf{R} matrix in the objective function.	132
3.8	Regulated flows using Scheme 2 with a non-zero \mathbf{R} matrix in the objective function.	132
3.9	Controlled water level using Scheme 1 with unknown demands from irrigation channels.	133
3.10	Regulated flows using Scheme 1 with unknown demands from irrigation channels.	134
3.11	Controlled water level using Scheme 2 with unknown demands from irrigation channels.	134
3.12	Regulated flows using Scheme 2 with unknown demands from irrigation channels.	135
3.13	Controlled water level in Lake Mulwala with a realistic forecast of demands from irrigation channels.	135
3.14	Regulated flows at Heywoods with a realistic forecast of demands from irrigation channels.	136
3.15	Inflows and outflows in the dataset.	137
3.16	Simulated flows at Bandiana and Peechelba.	137
3.17	Controlled water level using Scheme 1.	139
3.18	Regulated flows at Heywoods using Scheme 1.	140
3.19	Controlled water level using Scheme 2.	140
3.20	Regulated flows at Heywoods using Scheme 2.	140
3.21	Interpretation of the scenario theorem applied to scheme 3.	146
4.1	An example of loss distribution function.	153
4.2	An example of a probability density function of loss for a given decision.	154
4.3	A non-continuous loss distribution function and its normalised tails.	155
4.4	<i>Optimisation and Testing algorithm</i> (basic idea).	163
4.5	A plot of $\mathbb{F}(\frac{v}{N_T}, p)$ against p	171
4.6	<i>Optimisation and Testing algorithm</i>	177
4.7	Control of the output y	189
4.8	Alpha values computed by the algorithm (Section 4.3.2).	190
4.9	A pictorial description of violation sets.	193
5.1	Hume Reservoir to Lake Mulwala on the Murray River (plot not to scale).	200
5.2	Inflows and outflows in the dataset.	203
5.3	Measured and simulated flows at Bandiana and Peechelba.	203
5.4	Controlled water level in Lake Mulwala.	204
5.5	Regulated flows at Heywoods.	205
5.6	α values obtained in the <i>improving</i> phase (Problem (4.26)).	205
5.7	Controlled water level in Lake Mulwala with non-zero \mathbf{R} in the objective function.	206
5.8	Regulated flows at Heywoods with non-zero \mathbf{R} in the objective function.	207

5.9	α values obtained in the <i>improving</i> phase (Problem (4.26)), with non-zero \mathbf{R} in the objective function.	207
5.10	Controlled water level in Lake Mulwala.	209
5.11	Regulated flows at Heywoods.	209
5.12	α values obtained in the <i>improving</i> phase (Problem (4.26)).	210

List of Tables

2.1	SDP estimation from data with different range of variations.	46
2.2	CT and DT transfer functions corresponding to each inflow and outflow from Lake Mulwala.	49
2.3	SSPE for the water level simulations in Lake Mulwala.	56
2.4	CT and DT transfer functions corresponding to the inflows at Heywoods and Bandiana.	66
2.5	SSPE of water level simulations in Lake Mulwala and at Doctors Point. . .	71
2.6	Average SSPEs of 7 days simulation of water level in Lake Mulwala. . . .	73
2.7	CT transfer functions corresponding to each inflow and outflow from Lake Mulwala.	81
4.1	Binomial random variable V with its possible outcomes and probabilities.	170

This page intentionally left blank.

Chapter 1

Introduction

This thesis addresses the problem of improving water resource management. Water resources include rivers, irrigation channels, and dams and lakes along the rivers. Rivers play a major role in water distribution, and an efficient water distribution scheme along with a careful river control strategy can improve the water resource management. To effectively achieve this goal, the thesis considers three major topics,

1. Development of river models: The models are required to capture the river dynamics relevant for control, and they should be simple and easy to use for control design.
2. Design of an effective river control strategy: The strategy should be able to incorporate flow forecasts, irrigation demands and their uncertainties. Moreover, it should also be able to handle physical and environmental constraints.
3. Development of a flood mitigation mechanism: The river control strategy should assess the risk for a flood event and forecast potential upcoming floods, and it should take actions trying to avoid such an event.

1.1 Importance of research on water management

Water is a precious resource, and due to the rapid increase in world population and growth in agriculture sector, demand of water is growing day by day. It has led water to be a scarce resource, and the world can face severe water shortage in future if no remedies are made today. A public-private-civil society group (2030 Water Resources

Group, [5]) claims that if the current practices of water resource management continue, the world is projected to face a 40% water deficit by 2030 [6].

From a local perspective, water scarcity is a serious concern for Australia. The nation is considered the driest continent in the world. Moreover, agriculture is one of the major economic contributors, like for many other countries. These facts motivate the idea of properly addressing the problem of water resource management.

Currently, there is no shortage of water in Australia. The major concern is to develop an efficient way to supply water to minimise losses. As a rough measure, the water losses due to inefficient water distribution in irrigation channels in Australia are reported to be 30% of the supplied water [7]. The losses are mainly due to oversupply, leakage and seepage. To avoid serious water deficits in future, there is a need to explore new farming practices and better control strategies for rivers. That would require an interdisciplinary approach which includes agriculture science, engineering, ecology, hydrology etc. There has been several attempts made in this regard in the past, e.g. the Farms, Rivers and Markets (FRM) project. A part of the Rivers component of this project is reported in [8]. Some research work covered in this thesis is a part of Carlton Connect Initiative Fund (CCIF) project. This project is an initiative to form sustainable societies in Australia. It has four major areas: energy, water, food and urban future.

As mentioned earlier, rivers play a major role in water distribution as they form links between storages and irrigation channels to farms. The following points show some of the importance and challenges involved in river management and control.

- River management subject to constraints: Rivers supply water for farming, urban and industrial usage. However, there are several ecological and physical constraints imposed in order to preserve environment, river structure and its habitats. All such constraints have to be taken into account during the river operations.
- Efficient use of water: To meet environmental and irrigation demands, water should be released from the storages in an efficient way to avoid oversupply and/or undersupply.
- Uncertainties in unregulated flows: Water levels and flows in a river can be highly

affected by uncertainties in irrigation demands and unregulated inflows e.g. an inflow from a river with no controllable gates can be highly uncertain. Such uncertainties are required to be taken into account in the decision process.

- Flood avoidance: Apart from normal river operations which include irrigation, environmental, domestic and industrial supplies, there is a need to continuously assess the risk of a flood and to make amends in-advance to avoid or minimise possible damage.

Modelling and Control Systems Theory can play a vital role to ensure favourable river operations and management, and in this thesis we use the theory to demonstrate that. We consider the upper part of Murray River in Australia as a case study in this thesis. We describe the river in the next section.

1.2 Upper part of Murray River

Murray River is the longest river in Australia. Fig. 1.1 shows a sketch of the river stretch from Hume Reservoir to Lake Mulwala which has a river distance of 180 km and a straight line distance of 65 km. The river stretch are referred to as the 'upper part of Murray River'. The release from Hume is measured at Heywoods which is a kilometre downstream of the Hume Reservoir gates. Fig. 1.2 shows the gates and Fig. 1.3 shows the measuring station at Heywoods, where pressure sensors are installed to measure water level.

There are two unregulated rivers that flow into the river stretch (Fig. 1.1): Kiewa River and Ovens River. Kiewa River joins Murray River just downstream of Heywoods at Doctors Point as shown in Fig. 1.1. Inflow from the river is measured at Bandiana¹. Kiewa River is fed by several creeks and small water channels. The river also forms an ana-branch², which joins the main river from west. Ovens River joins Murray River just

¹Strictly speaking, flows are not measured but they are calculated from water level measurements using rating curves.

²Ana-branch is a branch of a river that diverts from the main river and rejoins the river further downstream.

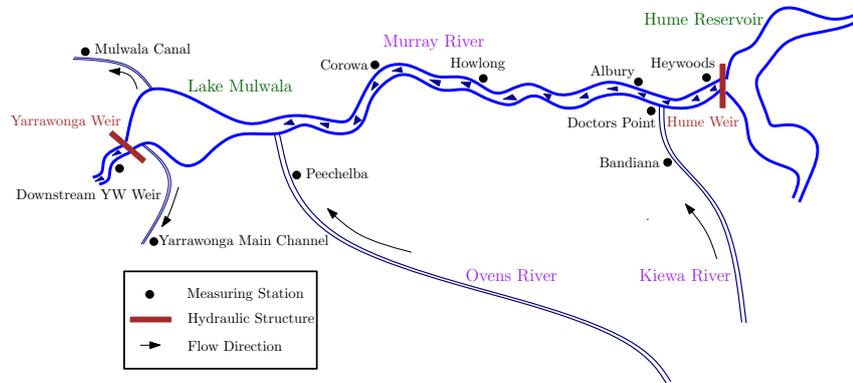


Figure 1.1: Hume Reservoir to Lake Mulwala on the Murray River (plot not to scale).



Figure 1.2: Hume Reservoir gates.



Figure 1.3: Measuring station at Heywoods.

upstream of Lake Mulwala as shown in Fig. 1.1. Inflow from the river is measured at Peechelba. The river is fed by several creeks and regulated and unregulated rivers.

There are several measuring stations on Murray River upstream of Lake Mulwala, such as Doctors Point, Albury, Howlong and Corowa. Lake Mulwala is the next reservoir on the Murray River after Hume Reservoir. Both reservoirs play important role in the regulation of Murray River. The stretch of Murray River from the point where Ovens River joins Murray to the upstream end of Lake Mulwala has ana-branches (not shown in Fig. 1.1). The downstream end of the lake has a weir; Yarrawonga Weir, which controls the release of water from the lake. Fig. 1.4 shows the Yarrawonga Weir. Two irrigation canals also originate from the lake; Yarrawonga Main Channel and Mulwala Canal. Fig. 1.5 shows the top view of Lake Mulwala, Murray River, Yarrawonga Weir and the two irrigation channels. The figure is taken from Google Maps.

Discharge and storage capacities in the upper part of Murray River are as follows (Murray Darling Basin Authority (MDBA) Canberra, personal communication, June 2013),

- Releases from Hume Reservoir can be made through 4 irrigation valves, 2 hydro



Figure 1.4: Yarrowonga Weir.

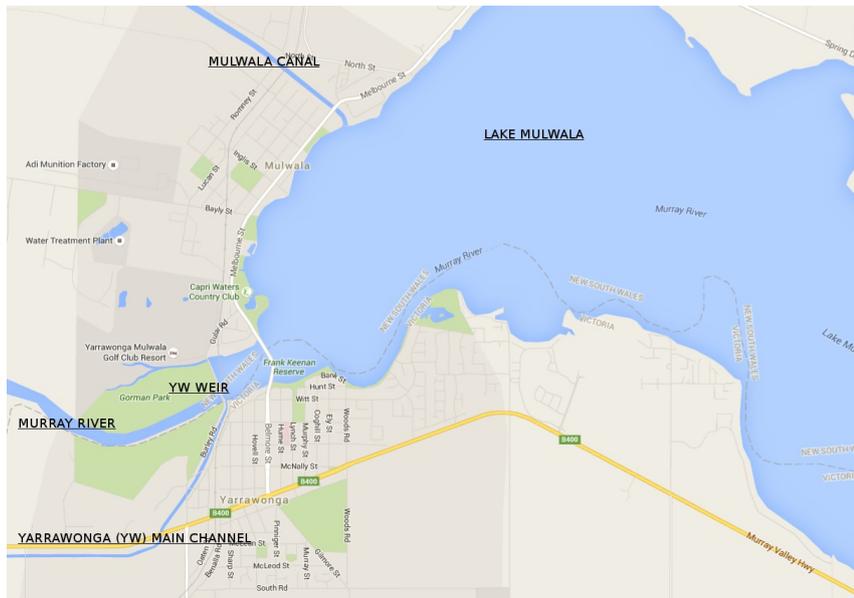


Figure 1.5: Top view of Lake Mulwala. Source: Google Maps, 2015.

turbines or 29 gates in the spillway.

- The maximum discharge from Hume is approximately 600,000 ML/day ($\approx 6,944 \text{ m}^3/\text{sec}$) at full supply level.
- Water level in Hume Reservoir is required to stay above 169.5 m AHD (meter Australian Height Datum, which is relative to sea level) for the hydro plant to operate. This level corresponds to 250 GL of water storage in the reservoir.
- Lake Mulwala has a storage capacity of 117.5 GL at full supply level, which is 124.9 mAHD.
- The main regulating structure at the downstream end of the lake has 8 steel vertical lift gates, and the northern regulating structure has 2 similar steel gates. The weir has two further gated regulating structures and two earthen embankments.
- Discharge capacity of Yarrawonga Main Channel is 3,170 ML/day ($\approx 37 \text{ m}^3/\text{sec}$).
- Discharge capacity of Mulwala Canal is 10,000 ML/day ($\approx 116 \text{ m}^3/\text{sec}$).

Major control objectives for the river stretch are,

- To maintain water level in Lake Mulwala between 124.65 mAHD and 124.9 mAHD.
- To timely provide water to the downstream end of Yarrawonga Weir, Yarrawonga Main Channel and Mulwala Canal.
- To avoid the fall of water level at Heywoods and Doctors Point to exceed 0.20 m/Day and 0.15 m/Day respectively.
- During normal operations the release from Hume Reservoir is only regulated to achieve the above control objectives and the flow release from Yarrawonga Weir is kept nearly constant. However, during flood operations both ends of the stretch are used to regulate water.

The aforementioned specifications and objectives are specific for the upper part of Murray River, however, the objectives are usually similar for most of the rivers around the

world. The upper part of Murray River is also vulnerable to floods, mainly due to the unregulated inflows from Kiewa and Ovens Rivers. It requires better control strategies that can accommodate uncertainties in flow forecasts and handle constraints, which are investigated in this thesis. The case study of Murray River covers most of the aspects we intend to study in this thesis. It is a natural extension to the study carried in [8], where Broken River was considered, which is a smaller river with few unregulated inflows.

1.3 Key steps to achieve better river management

Based on the above discussions, the thesis considers the following research questions to ensure better river management,

1. Which modelling technique is suitable for river control purposes?
2. What is a promising river control strategy?
3. How to ensure flood avoidance?

Each question is briefly discussed below.

1.3.1 Modelling

The literature provides both physical and data-based models for open channels³. In this thesis we use and build only data-based models.

In physical modelling the models are derived from the laws of Physics. The Saint Venant equations ([9]) can be used to describe river dynamics. The equations are non-linear Partial Differential Equations (PDEs), and they are derived from mass and momentum balances, for details see [9]. There is a large literature on the derivation and the numerical solutions of the Saint Venant equations, see e.g. [9, 10]. The boundary conditions of the PDEs are usually given by the flows at hydraulic structures, e.g. weirs and overshot gates [11]. The Saint Venant equations provide good insight into the system but they are complex to derive and require several physical parameters related to rivers

³Open channels have free surface e.g. irrigation channels and rivers.

which are sometimes not available. Moreover, they are non-linear, which make them difficult to use for control design.

On the other hand, data-based modelling or empirical modelling can be used because of widely available operational data from rivers, see e.g. [12–17]. Data-based modelling builds model from the operational data using system identification techniques [18, 19]. Models built from data are mostly simple and convenient to use for control design purposes, and they are known as ‘black-box models’. If some prior information is available, it can be accommodated in the model, and such models are called ‘grey-box models’. We seek grey-box models in this thesis. However, there are certain limitations of data-based modelling, e.g. the data should be suitable for system identification, i.e. it should have persistent excitations (see [18]), modelling highly depends on its intended use, for instance a river model obtained from low flow data would not be able to describe the river well during flooding, and also, travelling time of water varies with flows and it is possible that a data-based model with constant delays might not work well in such situations. However, there are a few works that present flow-dependent time delays in data-based models, e.g. see [20, 21].

In this thesis we have looked at the following problems related to river modelling,

- The literature provides different system identification methods, which are used for river modelling, we analyse them and compare their performances. We report the advantages and shortcomings of the identification methods in the context of rivers.
- The river used as the case study in this thesis is wider and larger with longer time delays between locations with regulating structures. The river is also fed by unregulated rivers, which are modelled separately.
- We briefly build non-linear river models and compare them with the linear models in terms of their descriptive performances.

The river modelling is mainly done for control purposes in this thesis.

1.3.2 Control design

The aim of river control is to keep water levels and flows in the river at required levels, while satisfying timely delivery to farms and irrigators and avoiding water wastage. It also includes satisfaction of physical and environmental constraints and consideration of the uncertainties associated with unregulated inflows and demands. Based on the above requirements, a river control problem can be defined as an optimisation problem with several objectives to be met and various constraints to be satisfied. In this thesis we have carried out the following steps to achieve the above control objectives,

- A Model Predictive Control (MPC, [22]) based strategy can be used to achieve river control. However, to ensure the incorporation of the unregulated flow forecasts and their uncertainties, we use a stochastic version of MPC, and define the optimisation problem as a Chance-Constrained optimisation Problem (CCP) ([23–26]), which is generally non-solvable. To find an approximate solution to a CCP, we use the scenario approach ([27–30]) and solve the optimisation problem in an MPC set-up.
- In the optimisation problem we optimise over affine control policies, rather than control values to ensure feedback of the available river information, to generate suitable control actions.
- Later in Chapter 3, it will be shown that the river optimisation problem can get infeasible, as the unregulated inflows can not be bounded from above in a reasonable way. In order to deal with that, we have proposed optimisation schemes to ensure the feasibility of the river control problem.

In this thesis we assess our control design through simulations on the historical data of the upper part of Murray River.

1.3.3 Flood avoidance

Apart from the normal control objectives mentioned in the previous subsection, it is also required to keep an eye on unregulated inflows to a river and to assess the flood risk.

However, during normal operations, we do not want to be overly cautious and to degrade a control action with unnecessary flood avoidance measures. In this thesis we have carried out the following steps to achieve these objectives,

- We formulate river control and flood avoidance problems together as a Multiple Chance-Constrained optimisation Problem (MCCP).
- We propose a computationally efficient algorithm to solve MCCPs. The algorithm is named as *Optimisation and Testing Algorithm*.
- The proposed algorithm not only achieves the control objectives, but also, it does not degrade control actions for unnecessary flood avoidance during normal conditions.

The upper part of Murray River has remained vulnerable to unregulated inflows. Murray River faced some major floods in 2003, 2010 and 2012 in the recent past. In all these events, Ovens River contributed significantly to the flood. In Chapter 5, we incorporate the forecast of the unregulated inflows from Kiewa and Ovens River in the control problem and demonstrate the flood avoidance in simulations.

1.4 Literature review

The literature covers a large amount of works on modelling and control of open water channels, such as rivers and irrigation channels. In this thesis, our main emphasis is on rivers, and in this section, we mention some major works relevant to rivers.

1.4.1 Review on river modelling

A river carries water from a source (e.g. lake, dam or some water reservoir) to a sink (e.g. farms, sea, ocean etc). To model this channel, we need to establish a relationship between water level or flow at the sink and flow release from the source.

As mentioned earlier, we aim to model rivers for control purposes, and so this review does not cover works on rainfall-runoff modelling and catchment modelling, however,

we briefly mention a few related works. Moreover, works on physical modelling are only briefly discussed, because we are not pursuing physical modelling either, for a detailed review see [8]. However, works on data-based (empirical) river modelling are discussed in some detail, as we use it in this thesis.

Physical modelling for control purposes

As mentioned earlier, it is difficult to use the *Saint Venant* (SV) equations [9, 31, 32] for control purposes in their original form, because of their complexity. The equations can be simplified and linearised, and this approach was first followed in [33], where the authors assumed uniform flows. However, the assumption of having uniform flows in rivers is not valid, because rivers experience backwater effects due to obstructions. These effects were then investigated in [34], where the authors split the dynamics into two parts, i.e. a part with uniform flows and a part where the flows are affected by the backwater effects. This work introduced the well known *Integrator Delay Model* (IDM), which is then widely used for control design, e.g. see [35, 36].

A few works in the literature used the SV equations in its non-linear form for control design, e.g. see [37, 38]. However, they did some other simplifications, e.g. viscous friction was considered to be zero and no lateral inflow was considered, for details see [9]. The authors extended the work in [39, 40]. In [39], the authors designed a control which could handle channels with steeper bottom slopes.

Litrico and his co-workers have made a major contribution to the SV equations based modelling of open channels for control design purposes, see e.g. [41–43]. They also made some simplifications in the SV equations, either by considering zero lateral flow, or negligible inertial term, or by linearisation, which transformed the equations to the *diffusive wave equations*. The linearisation of the diffusive wave equations are known as the *Hayami* model. The obtained models were validated against the operational data of Baïse River in France. They also introduced *Integrator Delay Zero* (IDZ) model in [44, 45], which addressed the issues related to the validity of the IDM model in frequency domain. X. Litrico and his co-workers said that at low frequencies the IDM model described a channel well, however, it did not remain ideal at high frequencies. Nonetheless, based

on the fact that the focus is mainly kept on the low frequency components for control design, the IDM model can still be considered as a valid choice for control purposes.

For a more detailed review on physical modelling of open channel flows, see [8] and the references therein.

Data-based modelling for control purposes

River operational data can be used to obtain data-based river models, and we have followed this approach in this thesis. There is a choice for the model structure to be black-box or grey-box [18]. The black-box models do not provide any physical insight to the system. On the other hand, the grey-box models incorporate prior knowledge about the system, and for water systems, we usually have quite a bit of prior information available, and thus we adopted this approach in this thesis.

The use of system identification to obtain models of open channels for control design is recent. The approach gained a lot of popularity after the contribution of Weyer and his co-workers. In [46], the author obtained a simple grey-box model for the Houghton Main Channel (HMC), Australia. The identification routine incorporated time delays and a relationship of gate heights and flows above the gate ([11]) in the model structure. The model turned out to be very close to a parametrised IDM, which is in agreement with [34]. The author preferred Output Error (OE)-type model structures, because an OE model provides better description of low frequency properties of a system, which is important for control purposes. In [47], the authors extended the work for different gate structures, the model was validated against the operational data from Coleambally Main Channel (CMC), Australia. The models were used for control design and the controllers were implemented on the irrigation channels, for details see [48–50]. Later on, the research group extended the identification methods to rivers, and in [8, 51], the authors described the modelling of Broken River in Australia, and they also designed centralised and decentralised controllers for the river, based on the identified models.

The literature also include the contribution of a few other works who used empirical modelling of rivers for control purposes. In [15], the authors used a grey-box model structure for Dalälven River in Sweden. The obtained model was also a parametrised IDM. A

time delay model of Sevier River in USA is presented in [13], and the obtained models were validated against measured data. A state space model of Red River in Vietnam is derived from operational data in [14], where the authors used only time delays as prior information. In [52], the authors considered the Non-linear Auto Regressive with Exogenous Input (NARX) model to describe river dynamics. However, the model was not validated, and no further justifications on the selection of a non-linear model structure were given, apart from the fact that the dynamics of rivers are non-linear.

Young and his co-workers have made major contributions in hydrological modelling, and their contributions also cover rainfall-runoff modelling, see e.g. [16, 17, 53] and the references therein. The authors developed the Data Based Mechanistic (DBM) approach and used it for hydrological modelling, where the non-linearities in the system were modelled as State Dependent Parameters (SDPs) [54]. This technique has been used by several other groups, e.g. to model the effect of snow-melt on river flow [55], to model a reservoir subject to precipitation [56] etc. In this thesis, we have used this technique to analyse the need of SDPs in a parameterised IDM.

1.4.2 Review on river control

The literature covers a wide range of works on the control of open channels. Both centralised and decentralised control strategies⁴ have been used for the control of rivers with single and multiple reaches over the past years. It is hard to cover all of them here, however, we mention some major works, along with the control strategies used therein. For a comprehensive list of references, see [37] and the references therein.

One of the mentionable works from 1980s is the control of a long river stretch [57]. In this work the authors considered a problem of maintaining water flows along the river at particular set-points. The authors used a switching control mechanism and PI controllers to fulfil the tasks. In [58], the authors developed an LQG controller for a dam-river system. The control objectives were again to maintain flow at a set-point and the assurance of on-time water delivery. The controller was not implemented on any real

⁴In the centralised configuration, the control actions of all nodes or reaches of a channel are governed by a central controller, however, in the decentralised controller every reach has its own local controller.

river. However, in [59], the authors extended the work to control a two-dam river system, where the authors used an H-infinity control.

With the popularity of Model Predictive Control (MPC), the researchers used MPC strategies to control rivers, because in MPC it is easy to handle constraints. In [15], the authors used an MPC set-up to control the Dalälven River in Sweden. Due to the presence of hydro-power stations, the river experiences large variations in water level and the control objective was to control the variations. The results showed that the MPC controller is suitable in this regard.

In [60], a centralised MPC strategy was used to control a four-reach stretch of Aare River in Switzerland. Again, the objectives were to maintain water level close to set-points and minimise the variation of water level due to the flow discharge through the turbine in the river. The variation in water level was used to be large, especially when the river was used for navigation and the locks were used to by-pass the hydro-power station, which introduced more variations and turbine wear. Setz and his co-workers extended the work in [60], and again used an MPC based strategy to control the river with different control objectives, see [61]. The authors considered soft constraints, and for this, they introduced slack variables in the criterion.

In [62], a centralised MPC and decentralised PID strategies were considered to control a three-reach Arrêt-Darré/Arros system in France. The control objectives included maintaining the river flows at several locations in the system at desired set-points. In [63], the authors used a centralised MPC scheme to control North Sea and the Amsterdam-Rhine Canals in the Netherlands. The control objective was to make the river navigable and minimise variations in water level. Also, the control objectives included minimisation of the energy consumption due to water pumping. In [64], the authors used a centralised MPC strategy to control four reaches of the Elbe River in Germany. The control objectives were to make the river more navigable by pumping water in the river but also minimising the power consumptions due to pumping. Similar objectives were met in [65] to control three reaches of the Moselle River in Germany.

A lot of research has gone into the computational burden of a centralised MPC strategy. Rivers are slow systems, and in general, it is expected that a centralised control

strategy can suit rivers. However, large number of reaches in a river might create a concern. In [66], the authors considered a decentralised MPC strategy to control a 35-reach river. The objectives were to minimise the variation of water level in the river in the presence of flow releases from hydro-power stations. Although no tests were conducted on a real river.

In [8, 51], both centralised and decentralised strategies were considered to control Broken River in Australia with multiple reaches. The authors used data-based models of the river. The decentralised control consisted of PI controllers and the centralised scheme was made up of MPC. The control system achieved substantial water savings and improved level of service to irrigators.

In [67], the author considered uncertainties in weather predictions, while designing a river control by using a tree-based MPC approach. They constructed uncertainty trees, based on the ensemble data and available information, and then the trees were used in the MPC based optimisation problems. The developed strategy was tested on the data of Rhone River in Switzerland and Lake Maggiore at the border between Italy and Switzerland. Later, the researchers from the same group used the tree-based MPC approach to manage a four reservoir system, which is upstream to the Seine River in France, and it was presented in [68]. The control objective was to manage the reservoirs to avoid occurrences of floods or droughts.

Unregulated in- and out-flows from a river carry uncertainty, and it is desirable if a control scheme can accommodate those uncertainties. To the best of author's knowledge, there has been a few such attempts made to design a river control which accommodates the uncertain behaviour, see e.g. [67–69], including the ones mentioned in the above paragraph.

1.4.3 Review on flood control

There are various works available in the literature that target flood risk mitigation. In the last 10-15 years, MPC has been mostly employed by the researchers to achieve such tasks. We mention a few such works below.

In [70], a centralised MPC strategy was used for the flood regulation in the Demer

River in Belgium. The authors constrained the water level in the MPC formulation to avoid river bank spilling. The authors also used the rainfall measurements as inputs in the MPC problem. They showed successful working of their strategy on the river data from the year 1998, when the river was hit by a huge flood due to a large amount of rainfall. Later in 2013, some of the authors of [70] published another paper [71], and proposed a strategy to control floods in a river system with reservoirs. In that strategy, the controller uses the buffer capacity of a reservoir in an optimal way when there is a risk of flooding. This strategy was applied to the regulation of Demer River in [72]. The simulation results on the past operational data showed that the strategy clearly outperformed the existing control scheme at that time. In 2015, the co-authors of [70] reconsidered the flood control problem in [73], where they used a combination of Genetic Algorithm (GA) [74] and MPC to achieve the control objectives.

The Wivenhoe Dam along the Brisbane River in Australia was built to safeguard Brisbane from flood. However, the dam was unable to prevent the flood event in 2011. In [75], the authors took the dam as a case study, and developed a flood mitigation strategy using MPC. The simulation results in [75] shows that the flood could be significantly reduced with their control strategy. Later in 2013, another strategy was proposed in [69]. The idea was to tackle the uncertain inflows by considering multiple river models to represent inflows ranging from low to high flows. Optimal model selection is done in a minimum mean square sense. The idea was successfully implemented on the operational data related to the flood event in 2011.

In [76], the authors used MPC to mitigate the flood risks in the Ebro River in Spain. To tackle floods, the river operators divert water to flood plains through gates. The control objective was to regulate the gates in an optimal manner to minimise flood risks. The authors showed simulation results to substantiate their claim.

An MPC based control strategy was also used in [77] where the authors proposed a control design to open or close different barriers to protect an area against floods. The Rhine-Meuse Delta in the Netherlands was considered as their case study. They used the term 'hybrid MPC' because the controller either opened or closed the barriers while the variables as water levels and flows were treated in continuous domain. The control

objective was to achieve a trade-off between minimising the cost of using the barriers and the water level exceeding the defined safety reference level. Later in 2015, the co-authors of [77] published another paper [78], in which they targeted flood control of a low-lying delta system and their case study was again the Rhine-Meuse Delta in the Netherlands. In this paper they used MPC and proposed a new ‘large time step’ scheme in order to save computational time, in which different control variables can have different control time steps. The simulation results in the paper were promising.

1.5 Thesis overview

The thesis consists of three major sections: river modelling, river control and flood avoidance. We briefly describe them before presenting the thesis organisation.

In this thesis we use empirical modelling to model the upper part of Murray River in Australia, and we pursue grey-box modelling. We investigate a few system identification methods like Prediction Error Method (PEM), Maximum Likelihood (ML) [18,79], continuous time system identification [80,81], Data-Based Mechanistic (DBM) modelling [53] and Subspace Identification Method (SIM) [19,35] to find river models. The purpose of the comparison is to investigate which system identification techniques provide better river description and ease in incorporating available prior information.

For river control, we propose a Stochastic MPC based approach that can deal with uncertainties. It involves formulating the problem as a chance-constrained optimisation problem [23–26]. However, a chance-constrained problem is non-convex in general, and is difficult to solve. There are a few alternative strategies available in the literature to find approximate solutions, see e.g. [27–30,82–85]. We select the scenario-based strategy [27–30,82] in this thesis, because it seems to be a promising choice as it is computationally tractable and it does not require any specific assumption on the nature of the disturbance.

For flood avoidance, we use a risk measure (Value at Risk [86]) to find flood risks associated with a particular control action, and then based on that, the controller decides to either modify the control action or not. We incorporate the risk measure in the Stochastic MPC control scheme, developed for normal river control operations, and if the risk

associated with a flood gets high, the controller modifies (improves) the control actions accordingly. Also, such a strategy prevents the control actions from getting too conservative during normal operations and avoids unnecessary flood risk mitigation.

In this thesis, we use the scenario based Stochastic MPC approach to control the upper part of Murray River in Australia, which is affected by unregulated inflows from Kiewa and Ovens Rivers. The thesis is organised as follows.

Chapter 2 focuses on the data-based modelling of rivers. We obtain models of the upper part of Murray River. In this chapter we use five different identification methods, consider multi-input single-output and multi-input multi-output models, and compare their performances. The ability of the identification methods to incorporate available prior information and the ease of identifying models are assessed.

Chapter 3 is dedicated to river control design. In this chapter we mathematically formulate the river control optimisation problem and use scenario-based Model Predictive Control as the control technique. We build schemes that ensure feasibility of the optimisation problems and we use them to control the upper part of Murray River and present simulation results.

Chapter 4 formulates a general control problem with an inclusion of risk mitigation as a Multiple Chance-Constrained optimisation Problem (M-CCP). The chapter proposes a computationally efficient algorithm: *Optimisation and Testing algorithm*, to find approximate solutions to M-CCPs.

Chapter 5 includes flood mitigation in the river control problem formulated in Chapter 3. The resulting optimisation problem is an M-CCP with two chance-constraints. We use the *Optimisation and Testing algorithm* to solve the problem and ensure flood mitigation with probabilistic guarantees. Simulations of the Murray River stretch is included in this chapter with favourable results.

Chapter 6 presents the conclusions and some areas for future works.

1.6 Major contributions of the thesis

The main contributions of the thesis are summarised as below.

- Data-based modelling of the variables of the upper part of Murray River. The variables include water level in Lake Mulwala and flow at Doctors Point.
- Comparison of different system identification methods in the context of river control.
- Formulation of a scenario-based Model Predictive Control (S-MPC) problem for rivers.
- Formulation of an optimisation scheme that gives feasibility assurance of the optimisation problems, involved in the S-MPC problem.
- Assessment of the proposed control strategy by implementing it on the upper part of Murray River in simulations.
- Formulation of a flood control scheme as a multiple chance-constrained optimisation problem with two chance-constraints.
- Proposition of a computationally efficient algorithm to find approximate solutions to multiple chance-constrained optimisation problems.
- Implementation of the proposed algorithm to flood data from the upper part of Murray River in simulations.

Chapter 2

System Identification Methods Applied to the Upper Part of Murray River

Due to the rapid development in sensors and information and communication technology, operational data from rivers are now often available, and system identification methods can be used to find models of large scale open water systems such as irrigation channels and rivers. The models obtained from system identification are simple ordinary differential or difference equations and are suitable for control design, prediction and fault detection purposes.

In this chapter we use several system identification techniques to find models of the upper part of Murray River in Australia (see the river description in Section 1.2). The methods we consider are Prediction Error Method (PEM), Maximum Likelihood (ML), continuous time system identification, Refined Instrumental Variable (RIV) method (used within the context of Data-Based Mechanistic (DBM) modelling) and Subspace Identification Method (SIM). Some of the methods have previously been applied to rivers, e.g. PEM was used in [12,51] & [13], SIM was used in [14], and DBM was applied in [15–17,53].

Important factors in system identification of rivers are the accuracy of the obtained model and its suitability for the intended purpose, the ability to incorporate prior information such as water level-flow relationships [11], the ability to handle multiple inputs and outputs, and the availability of easy to use software.

The above listed methods have their own advantages and drawbacks. It is relatively easy to incorporate prior information in PEM and ML based approaches [18] & [79]. The

Data-Based Mechanistic (DBM) approach has been successfully applied to many catchments and rivers (e.g. [16,17,53]), especially if rainfall-runoff effects need to be taken into account. Subspace identification [19] & [87] is well suited for Multi-Input, Multi-Output (MIMO) systems, and rivers are often MIMO with inflows from a number of tributaries, and we are often interested in modelling many flows and water levels along the river. Continuous time system identification e.g. [81] is also of interest since continuous time models are often preferred due to easy interpretations and usage. However, each method has drawbacks too, e.g. for optimisation based methods like PEM and ML, it can get computationally difficult to estimate a model with many parameters, and for subspace methods, it becomes challenging to incorporate available prior information if the number of outputs increases.

In this chapter we compare the above identification techniques, and in particular we consider (i) simulation performance on validation data, (ii) ability to incorporate available prior information, and (iii) ease in identifying models. The intended use of the models is to design controllers, which is considered in the proceeding chapters of this thesis.

The chapter is organised as follows. In Section 2.1 we describe the upper part of Murray River, the operational objectives and the available prior information. We also narrow down the phenomena to be modelled either due to lack of data or due to minor relevance for control. Furthermore we present the dataset used for identification and estimate key time delays in the system. In Section 2.2 we briefly discuss each identification method and apply them to Multi-Input Single-Output (MISO) models of the upper part of Murray River. Section 2.3 is dedicated to MIMO models. In Section 2.4 we compare the identified models and give some conclusions.

2.1 Prelude

In this section we discuss the operational objectives of the upper part of Murray River (Section 1.2) and describe the available prior information. Furthermore, we present the data used for system identification in this chapter.

For ease, we re-present the sketch of the upper part of Murray River in Fig. 2.1

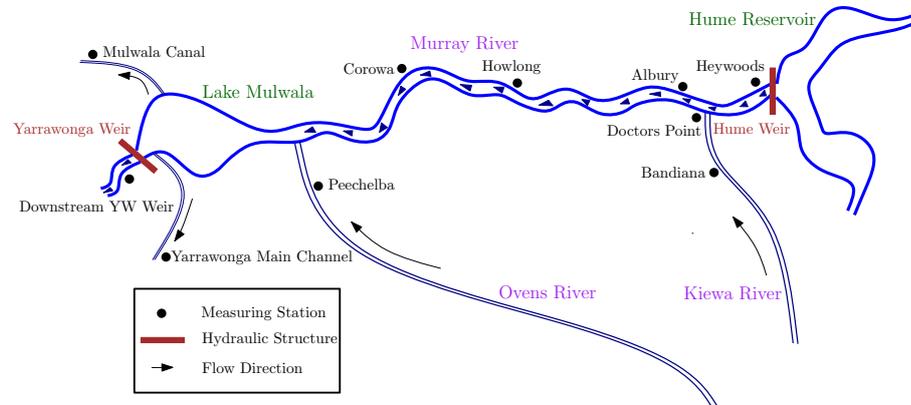


Figure 2.1: Hume Reservoir to Lake Mulwala on the Murray River (plot not to scale).

2.1.1 Operational objectives

The two main operational objectives for the river stretch (as described in Section 1.3.2) are

1. The water level in Lake Mulwala should be kept between 124.65 and 124.9 mAHD (meter Australian Height Datum, which is relative to sea level). This is required in order to facilitate gravity fed diversion of water into Mulwala Canal and Yarrowonga Main Channel, safe boating and recreational activities in the lake.
2. The rate of fall in the water level at Heywoods and Doctors Point should be less than 0.20 m/Day and 0.15 m/Day respectively, to avoid river bank slumping. Doctors Point is just downstream of Heywoods (see Fig. 2.1).

During normal operations the river stretch is controlled from Hume Reservoir only, and the release from Yarrowonga Weir is used to meet downstream operational objectives. However, during flood operations Yarrowonga Weir is also used to achieve the objectives listed above.

In this chapter we consider Multiple-Input, Single-Output (MISO) models and Multiple-Input, Multiple-Output (MIMO) models of the water level in Lake Mulwala and the flow at Doctors Points which are the two most important variables from an operational perspective. The purpose of the model is to use it for control design, and it is not intended for flood prediction. Control design is considered in the proceeding chapters of

this thesis.

2.1.2 Available prior information

The following prior information is available for the upper part of Murray River,

1. The direction of the flows in Murray River as indicated in Fig. 2.1.
2. Flow releases from a storage only affect downstream water levels, downstream flows and the immediate upstream water level in the storage.
3. Lakes and reservoirs can often be modelled by using a volume balance. E.g. for Lake Mulwala we can use the following approximate volume balance as a starting point,

$$\frac{dV_{LM}}{dt} = Q_H(t - \tau'_H) + Q_B(t - \tau'_B) + Q_P(t - \tau'_P) - Q_{DYW}(t) - Q_{YMC}(t) - Q_{MC}(t), \quad (2.1)$$

which simply says that the net change in volume of the water in the lake (V_{LM}) is equal to the sum of the inflows (Q) from Heywoods (H), Bandiana (B) and Peechelba (P) minus the sum of the outflows at (Downstream) Yarrawonga Weir (DYW), Yarrawonga Main Channel (YMC) and Mulwala Canal (MC). τ'_H, τ'_B and τ'_P are the time delays from Heywoods, Bandiana and Peechelba to the lake. We consider Eq. 2.1 as a starting point for selecting a model class in a system identification setting. However, as mentioned in the previous sub-section, water level in the lake is the important variable in the river stretch, and because of that we want the water level as the output rather than the volume. Therefore, assuming direct proportionality between the volume and the water level in the lake, we get

$$\frac{dy_{LM}}{dt} = c'_H Q_H(t - \tau'_H) + c'_B Q_B(t - \tau'_B) + c'_P Q_P(t - \tau'_P) - c'_{DYW} Q_{DYW}(t) - c'_{YMC} Q_{YMC}(t) - c'_{MC} Q_{MC}(t). \quad (2.2)$$

Additionally, we have introduced unknown parameters ($c'_H, c'_B, c'_P, c'_{DYW}, c'_{YMC}$ and c'_{MC}) in the model structure to encompass losses and gains due to leaks, evaporation, rain or surface water/ground water interactions. However, it should be

stressed that due to the assumption of the volume to water level proportionality, we get non-compatible units i.e. m and m^3/sec , rather than m^3 and m^3/sec , and Eq. 2.2 is not a volume balance any more. So it is difficult to physically interpret the unknown parameters and the losses and gains from model errors. However, the model errors will still be functions of these losses and gains. Also, we have not included any specific term for unaccounted-for losses and gains in Eq. 2.2. In that sense, one can argue that the models obtained using this model structure will be of limited use from a general hydrological modelling point of view, but it will be shown that they represent the dynamic behaviour of the selected river stretch well, from a control design perspective, which is the main purpose of modelling in this chapter.

We incorporate the aforementioned information in the river models. However, we do not consider some other prior information either to avoid modelling dynamics which are not important for control design or because the relevant measurements are not available. Such prior information include

1. Rainfall-runoff relationships for the unregulated rivers and Murray River itself. If rainfall measurements were available in the catchments of the Kiewa and Ovens Rivers, then the forecast models for the flows at Bandiana and Peechelba could be identified. However, we do not have the necessary rainfall data, but we have identified flow models at Bandiana and Peechelba from their upstream flows in Appendices 2.C & 2.D. Rainfall-runoff relationships have been incorporated in hydrological modelling in the literature, e.g. see [16,53,88].
2. Evaporation. It can be included in the volume balance for Lake Mulwala, but again there is a lack of data. Although, the net effect of evaporation is always a water loss, the variations within a day give rise to high frequency dynamics relative to the frequency range relevant for control.
3. Non-linearities due to anabranches¹. The upper part of Murray River has anabranches just upstream Lake Mulwala (not shown in Fig. 2.1) which most of

¹The branches that divert from the main river and rejoin the river further downstream.

the time are dry, but fill up during high flows. However, they are not significant for the flow regime we consider [89].

4. Relationship between water levels and flows at hydraulic structures. Such relationships are available in the literature [11], e.g. for weirs and overshot gates flows can be approximated by $Q(t) = ch^3(t)$, where Q is the flow, c is a gate/weir constant and h is the head above the gate/weir which is calculated as $h(t) = y(t) - p(t)$, where y is the water level upstream of the gate/weir and p is the gate/weir position. Such relations can be incorporated in river models, see e.g. [51] & [90]. However, we do not have data for the gate positions at Yarrawonga Weir, so we have not incorporated such relationships in Eq. 2.2.

2.1.3 Identification dataset

Not all operational river data are useful for identification purposes. Data can be non-informative from a system identification point of view for the following reasons

1. In some periods, there are strong correlations in flow data, e.g. the flow at Bandiana and Peechelba (Fig. 2.1) may follow same patterns. In such cases it is difficult to estimate the individual contributions from each flow.
2. Flow regimes can be very different from the flow regime a river model is supposed to capture.
3. There can be missing data and/or data from faulty sensors or inaccurate sensors which have not been calibrated for a long time.

For the upper part of Murray River we investigated 10 years of data from 2001 to 2010. In this work we focus on data with an operating range similar to a typical summer. This is because river management is usually more important in summer, when irrigation demands are high. As a rough estimate, flows at Heywoods and Yarrawonga Weir vary between 15,000 to 25,000 ML/day and 8,000 to 20,000 ML/day respectively during normal operations, inflows from the unregulated rivers; Kiewa and Ovens lie between 500

to 3,000 ML/day and 500 to 4,000 ML/day respectively. Moreover, the irrigation channels; Yarrawonga Main Channel and Mulwala Canal have releases up to 3,000 and 4,500 ML/day respectively during normal operations, and the water level in Lake Mulwala is between 124.65 to 124.9 mAHD under such conditions.

The identification dataset is shown in Fig. 2.2. The dataset is taken from the last three months of 2002, which corresponds to early summer and mid summer seasons in Australia. The sampling interval of the dataset is one hour. The top two graphs in Fig. 2.2 show all the inflows and the outflows in the river stretch and the bottom graph shows the water level in the lake.

The measurements of the water level in Lake Mulwala contain high frequency components. The blue curve in the bottom graph of Fig. 2.2 depicts this behaviour. Fig. 2.3 shows a plot of the magnitude of the Discrete Fourier Transform (DFT) of the water level measurements in 2002 on datasets from summer and winter. A peak occurs at the frequency $0.0417 \text{ (hour)}^{-1}$ in the summer dataset. The time period, $1/0.0417 = 23.98 \approx 24$ hours is most likely associated with evaporation as the peak is more prominent in the summer dataset. This peak was also observed in summer data from other years. We are mostly interested in the low frequency dynamics as for control purposes the high frequency components are not important, and hence the data can be low pass filtered before used for identification. For this purpose we used a second order zero-phase Butterworth filter with cut off frequency $0.015 \text{ (hour)}^{-1}$ for the Subspace Identification Method in Sections 2.2 & 2.3. Most of the other identification schemes used in this chapter are based on Output Error (OE)-type predictors which provide a good description of low frequency properties [18], and hence low pass filtering was not required for those methods (details in the following sections). The bottom graph of Fig. 2.2 shows both actual and filtered water levels.

Due to the large differences in the magnitudes of the input and output data some identification methods may run into numerical problems. To avoid such problems, the data was occasionally scaled prior to identification, and in the proceeding sections, the scaling is mentioned whenever applied to the identification dataset.

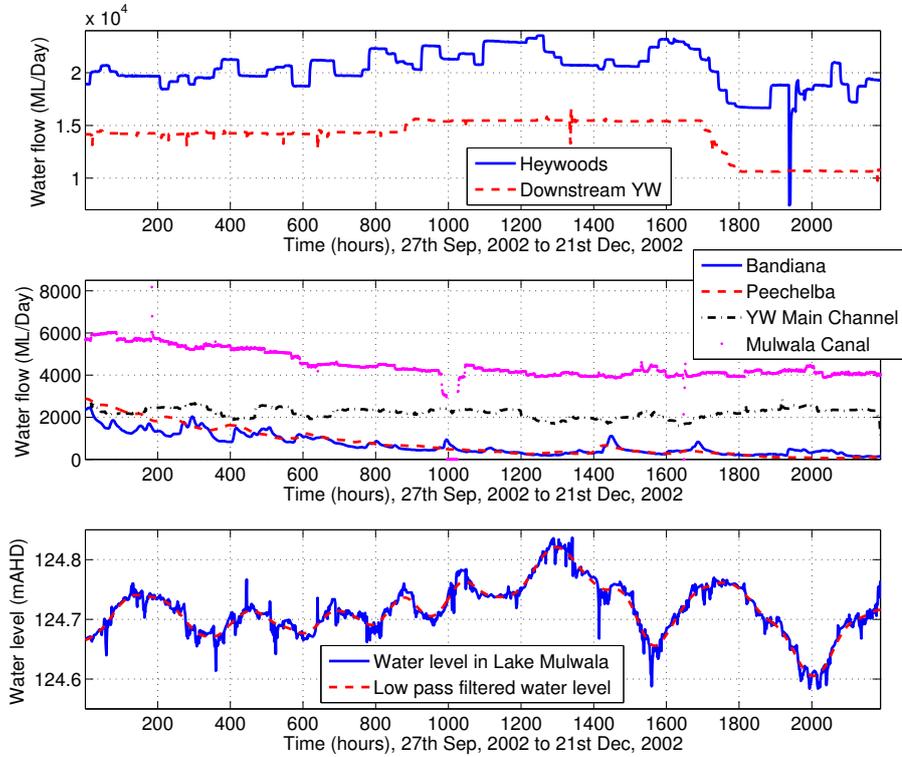


Figure 2.2: Identification dataset.

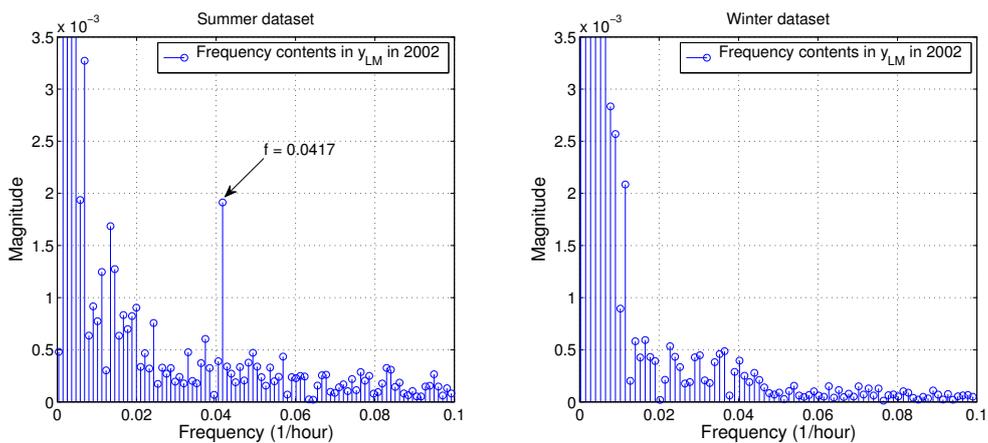


Figure 2.3: Magnitude of the DFT of the water level measurements at Lake Mulwala from summer and winter datasets.

2.1.4 Time delay estimation

Most river models contain time delays because flows and water levels are measured at different locations, and it takes time for the water to travel from one location to another. These time delays must be incorporated in a model. Time delay estimation is a common problem in all identification methods, and we chose to estimate them separately and treat them as a prior knowledge.

Many system identification methods e.g. Prediction Error Methods (PEM) and Maximum Likelihood (ML) methods ([18] & [79]) are based on solving an optimisation problem to identify the unknown parameters in a model. Time delays—if considered as unknown parameters—can also be estimated by solving the optimisation problem. However, the estimates obtained that way can be unreliable and not in agreement with the physical reality. One reason for this is that the flows in a river can be approximately constant for long periods of time which makes it difficult to distinguish between different time delays. Subspace Identification Methods (SIM) [19] & [87], can estimate the order of a system, and thus the time delays, by analysing the singular values of certain subspaces related to the system's state space model. However, SIM often fails to detect the order of systems with time delays. The reason for this is that the time delays introduce zero eigenvalues and the other eigenvalues of the system are likely to be far away from zero. As a consequence the overall system becomes stiff, and identifying such a system is difficult [19].

Constant time delays

In the literature there are a number of methods available for estimating time delays in open channels and rivers, see e.g. [12, 14, 20, 50, 53, 91]. Most of them are based on cross-correlation methods which obtain estimates by computing the cross-correlation between the upstream and downstream flows, and we have followed this approach here since it usually gives reliable estimates. The literature also provides several methods for constant time delay estimation of linear systems like time domain approximation methods, frequency domain approximation methods, Laguerre domain approximation methods,

explicit time delay parameter method, area and moment method (for details see [92] and the references therein).

Varying time delays

Time delay vary with inflow, and it decreases with an increase in flow. However, it depends on the physical geometry of the river as well. If an inflow keeps on increasing, then a part of the river bed or a branch that is normally dry may start filling up with water, and that increases the time delay. There are a few methods available in the literature for incorporation of time delay as a function of inflow (see [20]). In the extended cross-correlation coefficient method (discussed in [20] & [21]) both upstream and downstream flows are divided into bands corresponding to the percentages (10%, 20%, . . . , 100%) of the highest flows. Different events in data, especially flow peaks, can be used to calculate the corresponding time delays. The trend in the variation of time delay with an inflow can then be formulated as a linear regression problem. Instead of assuming a linear relation between the bands, an exponential function can also be used, e.g. $\tau(Q_{in}) = A + Be^{-cQ_{in}}$ [20], where A , B and c are unknown parameters. The identification dataset is divided into two subsets; primary and auxiliary. The primary dataset is used to calibrate the parameters of the model, and the auxiliary dataset is used to estimate the parameters associated with the time delay. However, varying time delays are of less interest here since they remain nearly constant in the low flow regime during summer which we are mainly interested in.

2.1.5 Estimating time delays: τ_H , τ_B and τ_P

One problem with methods based on cross-correlation is that they fail when the flows at the upstream and downstream ends are regulated independently. In such cases we can cross-correlate the upstream flow with an intermediate unregulated flow and extrapolate the estimated time delay to the downstream end. The extrapolated time delay can be used to narrow down the range of time delays to be considered in an optimisation based system identification method. We use such an approach to estimate the time delays from

Heywoods to Yarrawonga Weir (τ_H), Bandiana to Yarrawonga Weir (τ_B) and Peechelba to Yarrawonga Weir (τ_P).

The release from Yarrawonga Weir is independently regulated. Hence there is not much natural correlation between the flows at Heywoods, Bandiana and Peechelba and the flow over Yarrawonga Weir, and it is difficult to find the time delays using a cross-correlation method. However, the flow at Corowa is correlated with the flows at Heywoods and Bandiana. To begin with, we estimate the time delays from Heywoods to Corowa ($\tau_{H \rightarrow C}$) and Bandiana to Corowa ($\tau_{B \rightarrow C}$) and then use this information to narrow down the range of time delays to Yarrawonga Weir. The estimate of $\tau_{H \rightarrow C}$ is given by

$$\hat{\tau}_{H \rightarrow C}(\delta) = \arg \max_{\delta} \frac{1}{N - \delta} \frac{\sum_{n=\delta+1}^N [Q_H(n - \delta) - \bar{Q}_H][Q_C(n) - \bar{Q}_C]}{\sigma_{Q_H} \sigma_{Q_C}}, \quad (2.3)$$

where $\hat{\tau}_{H \rightarrow C}$ is the time delay estimate, N is the number of data samples, $\bar{Q}_i = \frac{1}{N} \sum_{n=1}^N Q_i(n)$, $i = H, C$ are the average flows and $\sigma_{Q_i} = \sqrt{\frac{1}{N} \sum_{n=1}^N (Q_i(n) - \bar{Q}_i)^2}$, $i = H, C$ are the standard deviations of the flows. We used various hourly sampled datasets from 2001 to 2010 and calculated the time delays using Eq. 2.3. We observed differences in the estimate of the time delay between high and low flow regimes. It varied from 44 to 55 hours and on average it was $\hat{\tau}_{H \rightarrow C} = 51$ hours. Fig. 2.4 shows an examples of flow plots with corresponding cross-correlation coefficient. The maximum value gives the time delay estimates from Heywoods to Corowa. Using the same approach we obtained the estimate $\hat{\tau}_{B \rightarrow C} = 52$ hours for the time delay between Bandiana and Corowa. Fig. 2.5 shows an example of the flow plots with the corresponding cross-correlation coefficient. The flow at Bandiana which is shown with a blue curve in Fig. 2.5 reveals a periodic component, especially in the first half of the dataset. This is most likely due to a 24 hour release cycle from an upstream hydro electric power plant. To find τ_H and τ_B we need to add the travelling time from Corowa to Yarrawonga Weir to $\hat{\tau}_{H \rightarrow C}$ and $\hat{\tau}_{B \rightarrow C}$ respectively. Based on the distances and elevations we estimated that the time delays τ_H and τ_B should be in the range 60 to 80 hours and τ_P in the range 6 to 18 hours.

The estimates are then further refined using optimisation based system identification methods in the following sections.

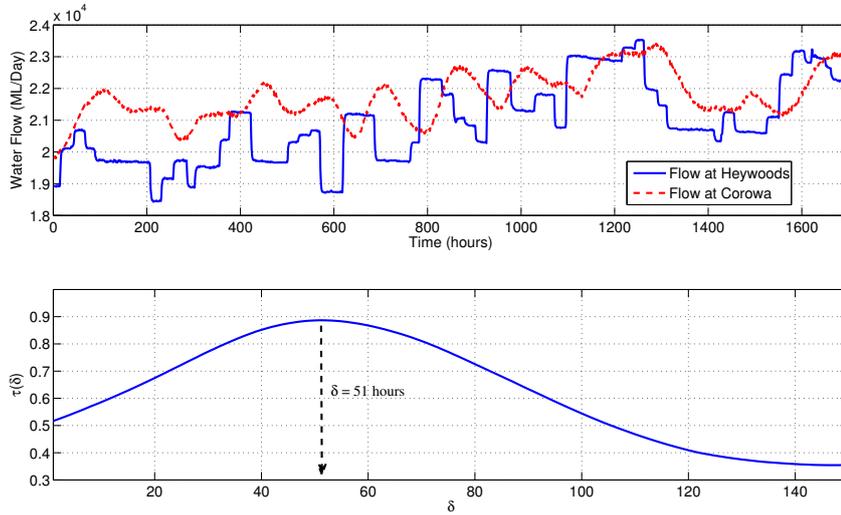


Figure 2.4: Cross-correlation coefficient, the maximum value gives the estimate of the time delay from Heywoods to Corowa (data is from 27th Sep, 2002 to 3rd Dec, 2002).

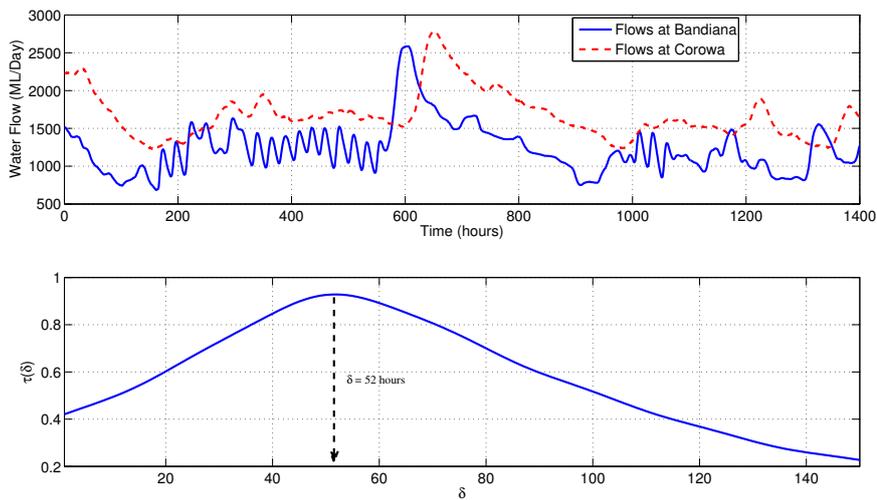


Figure 2.5: Cross-correlation coefficient, the maximum value gives the estimate of the time delay from Bandiana to Corowa (data is from 22nd July, 2007 to 20th Sep, 2007).

2.1.6 Model orders and noise models

In this thesis we only consider first order models. Higher order models have also been tried but there were little or no improvements, and hence we limit the presentation here to first order models with constant time delays. Also, we do not consider noise models because our aim is to obtain models for control purposes.

2.2 Modelling water level in Lake Mulwala

In this section we describe different identification methods and use them to identify MISO models of the water level in Lake Mulwala followed by a comparison and a discussion of their simulation performance against validation data. We implemented our own identification routines for all methods except for the continuous time RIV method and for the State Dependent Parameter (SDP) estimation, where we used the CONTSID[2] and CAPTAIN [1] toolboxes respectively in Matlab.

2.2.1 Prediction Error Method (PEM)

In PEM [18] & [79] we select a model structure $\mathcal{M}(\theta)$ parametrised by a parameter vector θ , and we seek a “good” model by minimizing a non-negative function of the prediction errors $\varepsilon(t, \theta)$ with respect to θ . The prediction errors are given by

$$\varepsilon(t, \theta) = y(t) - \hat{y}(t|\theta), \quad (2.4)$$

where $y(t)$ is the observed output at time t and $\hat{y}(t|\theta)$ is the predicted output based on the model parameters θ .

Model structure and predictor

In PEM it is easy to incorporate prior information in the model structure. We use Eq. 2.2 as a model structure for the water level in Lake Mulwala. Using an Euler approximation

for the derivative, we obtain the following discrete time model,

$$y_{LM}(n+1) = y_{LM}(n) + c_H Q_H(n - \tau_H) + c_B Q_B(n - \tau_B) + c_P Q_P(n - \tau_P) + c_{DYW} Q_{DYW}(n) + c_{YMC} Q_{YMC}(n) + c_{MC} Q_{MC}(n). \quad (2.5)$$

Here $\theta = [c_H \ c_B \ c_P \ c_{DYW} \ c_{YMC} \ c_{MC}]^\top$ and $\tau = [\tau_H \ \tau_B \ \tau_P]^\top$ are the unknown parameters. We expect c_H , c_B and c_P to be positive and c_{DYW} , c_{YMC} and c_{MC} to be negative as they correspond to in- and out-flows respectively. We use Eq. 2.5 to define an Output Error (OE) type predictor as

$$\hat{y}_{LM}(n+1|\theta, \tau) = \hat{y}_{LM}(n|\theta, \tau) + c_H Q_H(n - \tau_H) + c_B Q_B(n - \tau_B) + c_P Q_P(n - \tau_P) + c_{DYW} Q_{DYW}(n) + c_{YMC} Q_{YMC}(n) + c_{MC} Q_{MC}(n). \quad (2.6)$$

We prefer an OE type predictor rather than an ARX type predictor which uses $y_{LM}(n)$ on the right hand side of Eq. 2.6 instead of $\hat{y}_{LM}(n|\theta, \tau)$, since models obtained using OE predictors provide a better description of the low frequency properties [18] and for control purposes we are more interested in such descriptions. The parameters θ and τ in the OE model can be estimated using a quadratic criterion function [18] & [79], i.e.

$$(\hat{\theta}, \hat{\tau}) = \arg \min_{\theta, \tau} \frac{1}{N - \tau_{\max}} \sum_{n=\tau_{\max}+1}^N (y_{LM}(n) - \hat{y}_{LM}(n|\theta, \tau))^2, \quad (2.7)$$

where, $\tau_{\max} = \max(\tau_H, \tau_B, \tau_P)$. We can use the estimates of the time delays obtained in Appendix 2.1.5 and modify the optimisation problem in Eq. 2.7. A PEM estimate for θ and τ can then be found by solving,

$$(\hat{\theta}, \hat{\tau}) = \arg \min_{\substack{\theta, \\ \tau_H, \tau_B = \{60, 61, \dots, 80\}, \\ \tau_P = \{6, 7, \dots, 18\}}} \frac{1}{N - \tau_{\max}} \sum_{n=\tau_{\max}+1}^N (y_{LM}(n) - \hat{y}_{LM}(n|\theta, \tau))^2. \quad (2.8)$$

Eq. 2.8 defines a Mixed Integer Program which in general is difficult to solve. However, due to a small number of possible values of the delays, we can solve Eq. 2.8 by considering all possible combinations, and in each combination, τ_{\max} will be calculated as $\max(\tau_H, \tau_B, \tau_P)$.

Identified model and results on validation data

An OE model was identified using the dataset in Fig. 2.2. The smallest value of the identification criterion (Eq. 2.8) was achieved with time delays; $\tau_H = 70$, $\tau_B = 71$ and

$\tau_P = 16$ hours, and the parameters $\hat{\theta} = [6.20 \times 10^{-7} \quad 9.63 \times 10^{-7} \quad 5.89 \times 10^{-7} \quad -6.13 \times 10^{-7} \quad -9.73 \times 10^{-7} \quad -6.67 \times 10^{-7}]$. The first three parameters correspond to the inflows at Heywoods, Bandiana and Peechelba and the remaining parameters correspond to the outflows at Yarrawonga Weir, Yarrawonga Main Channel and Mulwala Canal. The signs of the identified parameters are in agreement with inflows and outflows.

In Fig. 2.6 the model is simulated against data where there are small variations in the actual water level in Lake Mulwala. In Fig. 2.7 the model is simulated against data where the water level has relatively larger variations. The corresponding in- and outflows of the four data are shown in Appendix 2.A.1. The simulation results show that the model picks up the trends very well, especially in Fig. 2.7, where the simulations are carried out for 1200 hours with convincing performance. The predictor model in Eq. 2.6 contains an integrator, and thus the simulated output is expected to drift. This is visible after around 350 hours (2 weeks) in the second plot of Fig. 2.6. A negative offset occurred and is carried till the end of the simulation. However, such errors can be avoided by reinitialising the water level at appropriate times. This is the strategy followed in the receding horizon approach in Model Predictive Control (MPC) [22], where the water level is initialised at each time step based on the measurements. Currently, the river operators are considering 90 to 170 hours (roughly 4 to 7 days) forecasts for this reach when they are planning the operations, and from the simulation results in Figs. 2.6 & 2.7, we can see that the identified model is suitable for simulations up to such horizon lengths, because the simulation lengths in the figures are much more than that with a single initialisation. The estimated time delays ($\tau_H = 70$, $\tau_B = 71$ and $\tau_P = 16$ hours) are in agreement with the experience of the river operators. However, slight differences in the time delays, e.g. up to 3-4 hours do not affect the performance of the identified model much. To demonstrate this, simulations of water level in Lake Mulwala with slight differences in the time delay from Heywoods (τ_H), using the identified model, are shown in Appendix 2.A.2.

In PEM it is easy to incorporate available prior information about non-linearities in the model structure. If the position of a gate is available along with the water level upstream of that gate, it can be used to describe the outflow by empirically derived relationships [11] discussed in Section 2.1.2. Such inclusions can improve the model per-

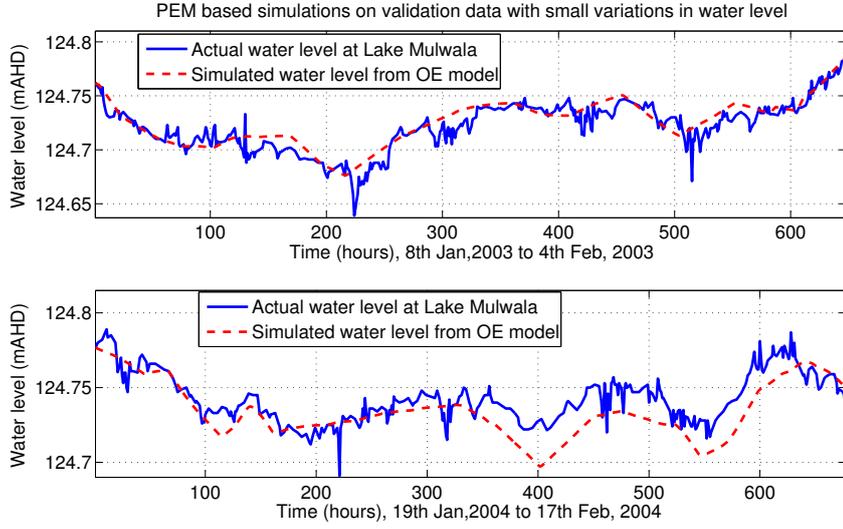


Figure 2.6: Simulation results on validation data with small variations in the water level in Lake Mulwala.

formance. Unfortunately, no information about the gate positions at Hume Dam and Yarrawonga Weir is stored so it could not be used here. However, the procedure for including this information is discussed in [51] & [90], and it has also been used for models of irrigation channels e.g. see [46], [50] and the references therein.

2.2.2 Maximum Likelihood (ML)

The ML approach [18] & [79] is based on statistical arguments, and in this section we consider this approach to identify a model of the water level in Lake Mulwala. We consider a noise free model of the water level based on the model structure in Eq. 2.5,

$$\begin{aligned}
 x_{LM}(n+1|\theta, \tau) &= x_{LM}(n|\theta, \tau) + c_H Q_H(n - \tau_H) + c_B Q_B(n - \tau_B) \\
 &+ c_P Q_P(n - \tau_P) + c_{DYW} Q_{DYW}(n) + c_{YMC} Q_{YMC}(n) + c_{MC} Q_{MC}(n).
 \end{aligned} \tag{2.9}$$

We further assume that the measured water level y_{LM} is $x_{LM}(n|\theta, \tau)$ corrupted by noise i.e.

$$y_{LM}(n) = x_{LM}(n|\theta, \tau) + e(n),$$

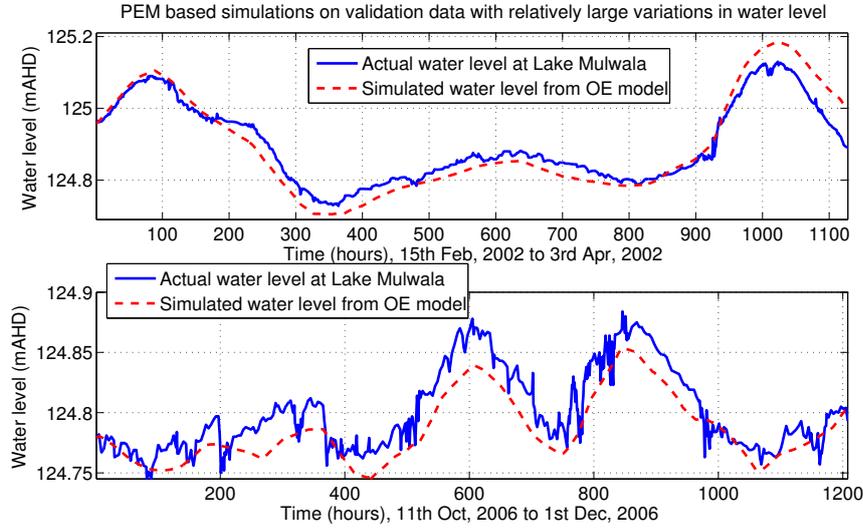


Figure 2.7: Simulation results on validation data with relatively large variations in the water level in Lake Mulwala.

where $\{e(n)\}$ is a sequence of independent and identically distributed (i.i.d), normal, zero-mean random variables with variance σ_e^2 . The prediction errors are given by

$$\varepsilon_{LM}(n|\theta, \tau) = y_{LM}(n) - x_{LM}(n|\theta, \tau), \quad \text{for } n = 1, 2, \dots, N,$$

where N is the number of identification data points. Using the above assumptions the density function is given by

$$f_{\varepsilon, n}(\varepsilon_{LM}(n)|\theta, \tau) \sim \mathcal{N}(0, \sigma_e^2) = \frac{1}{\sqrt{2\pi\sigma_e^2}} e^{-\frac{1}{2\sigma_e^2}(\varepsilon_{LM}(n|\theta, \tau))^2}, \quad (2.10)$$

from which it follows that the density function of $y_{LM}(n)$ is given by

$$f_{y, n}(y_{LM}(n)|\theta, \tau) \sim \mathcal{N}(x_{LM}(n|\theta, \tau), \sigma_e^2).$$

Due to the assumption of independence, the likelihood function is given by

$$\mathcal{L}(\theta, \tau|y_{LM}(1), y_{LM}(2), \dots, y_{LM}(N)) = \prod_{n=1}^N f_{y, n}(y_{LM}(n)|\theta, \tau),$$

$$\sim \prod_{n=1}^N \mathcal{N}(x_{LM}(n|\theta, \tau), \sigma_\varepsilon^2). \quad (2.11)$$

To estimate $\theta = [c_H \ c_B \ c_P \ c_{DYW} \ c_{YMC} \ c_{MC}]^\top$ and $\tau = [\tau_H \ \tau_B \ \tau_P]^\top$ in Eqs. 2.6 & 2.10, we need to maximise the likelihood function (Eq. 2.11) or equivalently minimise the negative log of the likelihood function, which is given by

$$-\log \mathcal{L} = \frac{N}{2} \log 2\pi + \frac{N}{2} \log \sigma_\varepsilon^2 + \frac{1}{\sigma_\varepsilon^2} \left(\sum_{n=1}^N (y_{LM}(n) - x_{LM}(n|\theta))^2 \right), \quad (2.12)$$

which is equivalent to the optimisation problems in Eqs. 2.7 & 2.8 obtained in the PEM approach. Thus, we get the same estimates of θ .

2.2.3 Continuous-Time (CT) identification

There are two major approaches to CT system identification.

1. The indirect approach in which a discrete time (DT) model is identified from the observed data and converted to a CT model.
2. The direct approach in which a CT model is identified directly from the observed data [80] & [81].

In this section we use the direct method. We consider the CT volume balance in Eq. 2.2 and obtain a model structure for the noise-free water level x_{LM}

$$\begin{aligned} \frac{dx_{LM}(t)}{dt} &= c_H Q_H(t - \tau'_H) + c_B Q_B(t - \tau'_B) + c_P Q_P(t - \tau'_P) \\ &+ c_{DYW} Q_{DYW}(t) + c_{YMC} Q_{YMC}(t) + c_{MC} Q_{MC}(t). \end{aligned} \quad (2.13)$$

We integrate Eq. 2.13 from an initial time t_0 to time t and get

$$\begin{aligned} \int_{t_0}^t \frac{dx_{LM}(\bar{t})}{d\bar{t}} d\bar{t} &= x_{LM}(t) - x_{LM}(t_0) = c_H \int_{t_0}^t Q_H(\bar{t} - \tau'_H) d\bar{t} + c_B \int_{t_0}^t Q_B(\bar{t} - \tau'_B) d\bar{t} \\ &+ c_P \int_{t_0}^t Q_P(\bar{t} - \tau'_P) d\bar{t} + c_{DYW} \int_{t_0}^t Q_{DYW}(\bar{t}) d\bar{t} \\ &+ c_{YMC} \int_{t_0}^t Q_{YMC}(\bar{t}) d\bar{t} + c_{MC} \int_{t_0}^t Q_{MC}(\bar{t}) d\bar{t}. \end{aligned} \quad (2.14)$$

We define a CT OE-type predictor from the above equation,

$$\begin{aligned} \hat{y}_{LM}(t|\theta, \tau) = & \hat{y}_{LM}(t_0|\theta, \tau) + c_H \int_{t_0}^t Q_H(\bar{t} - \tau'_H) d\bar{t} + c_B \int_{t_0}^t Q_B(\bar{t} - \tau'_B) d\bar{t} \\ & + c_P \int_{t_0}^t Q_P(\bar{t} - \tau'_P) d\bar{t} + c_{DYW} \int_{t_0}^t Q_{DYW}(\bar{t}) d\bar{t} \\ & + c_{YMC} \int_{t_0}^t Q_{YMC}(\bar{t}) d\bar{t} + c_{MC} \int_{t_0}^t Q_{MC}(\bar{t}) d\bar{t}, \end{aligned} \quad (2.15)$$

where $\theta = [c_H \ c_B \ c_P \ c_{DYW} \ c_{YMC} \ c_{MC}]^T$ and $\tau = [\tau'_H \ \tau'_B \ \tau'_P]^T$ are the unknown parameters.

We use the same hourly sampled dataset for identification as used for PEM (see Fig. 2.2). As the regulated flows are almost constant between samples, a zero order hold (zoh)-interpolation is used in order to obtain continuous data for the flows at Heywoods (H), (Downstream) Yarrawonga Weir (DYW), Yarrawonga Main Channel (YMC) and Mulwala Canal (MC), while a first order hold (foh)-interpolation is used for the unregulated flows at Bandiana (B) and Peechelba (P) where there are larger flow variations between samples. That is for $X = H, DYW, YMC$ & MC , and for $\bar{t} \in [t_0, t]$ we have $Q_X(\bar{t}) = Q_X(t_0)$ in Eq. 2.15, and for flows at $Y = B$ & P we have $Q_Y(\bar{t}) = Q_Y(t_0) + \frac{(\bar{t}-t_0)}{(t-t_0)}(Q_Y(t) - Q_Y(t_0))$. For the hourly sampled dataset we have sampling period $T = t - t_0 = 1$, and thus the predictor becomes

$$\begin{aligned} \hat{y}_{LM}(t|\theta, \tau) = & \hat{y}_{LM}(t-1|\theta, \tau) \\ & + c_H Q_H(t-1 - \tau'_H) + \frac{c_B}{2}(Q_B(t - \tau'_B) + Q_B(t-1 - \tau'_B)) + \frac{c_P}{2}(Q_P(t - \tau'_P) \\ & + Q_P(t-1 - \tau'_P)) + c_{DYW} Q_{DYW}(t-1) + c_{YMC} Q_{YMC}(t-1) + c_{MC} Q_{MC}(t-1), \end{aligned} \quad (2.16)$$

Compared to the model used in PEM (Eq. 2.6) there are now two delayed values of the flows at Bandiana and Peechelba in the model, although the number of parameters remain the same. For simplicity, we used the same time delays as before i.e. $\tau'_H = \tau_H = 70$, $\tau'_B = \tau_B = 71$ and $\tau'_P = \tau_P = 16$. We used the identification dataset in Fig. 2.2 and solved the following optimisation problem

$$\hat{\theta}_{CT} = \arg \min_{\theta} \frac{1}{N} \sum_{t=1}^N (y_{LM}(t) - \hat{y}_{LM}(t|\theta))^2. \quad (2.17)$$

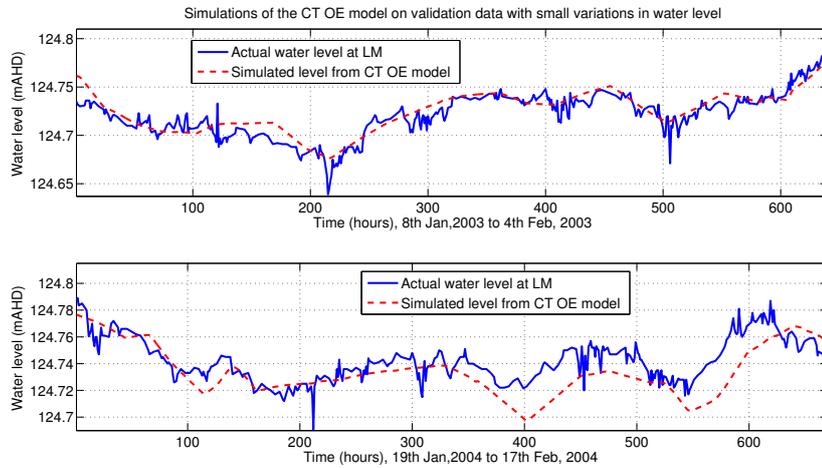


Figure 2.8: Simulation results on validation data with small variations in the water level in Lake Mulwala.

We got $\hat{\theta}_{CT} = [6.20 \times 10^{-7} \quad 9.61 \times 10^{-7} \quad 5.93 \times 10^{-7} \quad -6.14 \times 10^{-7} \quad -9.74 \times 10^{-7} \quad -6.68 \times 10^{-7}]$. The signs are in agreement with the inflows and outflows, and the estimates are almost the same as what we got with PEM in Section 2.2.1, which was expected since the flows are in most instances slowly varying. Figs. 2.8 and 2.9 show the performance of the identified OE model on validation data with small and large variations respectively. The model picks the trend very well and the results are similar to PEM.

2.2.4 Data-Based Mechanistic (DBM) approach

In this section we apply the DBM approach. The approach has been widely used in hydrological and environmental modelling for many years, see e.g. [16, 17, 53]. Within the context of this approach, one can identify static non-linearities in a system, which has been done in many works in the literature, specifically in rainfall-runoff modelling and in flood forecasting [17] & [93]. In this section, we first identify State Dependent Parameters (SDP) in a model of the water level in Lake Mulwala, and we have used the CAPTAIN toolbox [1] for this purpose. Later, we identify a CT OE model of the water level using a Simplified Refined Instrumental Variable method for Continuous time systems (SRIVC), within the context of DBM modelling ([80], [94]). There are two toolboxes that support this method in Matlab; CAPTAIN [1] and CONTSID [2], and we have used the CONTSID

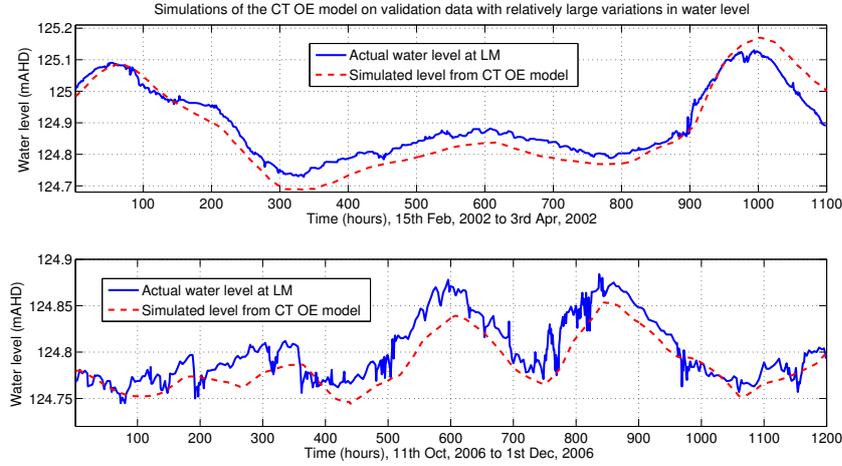


Figure 2.9: Simulation results on validation data with relatively large variations in the water level in Lake Mulwala.

toolbox in our work.

State Dependent Parameter (SDP) estimation

The idea of SDP estimation is to consider the parameters of a model to be functions of past inputs and outputs. One possibility is to let the parameters depend on their corresponding flows i.e.

$$\begin{aligned}
 y_{LM,n+1} = & y_{LM,n} + c_H(Q_{H,n-\tau_H})Q_{H,n-\tau_H} + c_B(Q_{B,n-\tau_B})Q_{B,n-\tau_B} + c_P(Q_{P,n-\tau_P})Q_{P,n-\tau_P} \\
 & + c_{DYW}(Q_{DYW,n})Q_{DYW,n} + c_{YMC}(Q_{YMC,n})Q_{YMC,n} + c_{MC}(Q_{MC,n})Q_{MC,n}, \quad (2.18)
 \end{aligned}$$

where subscripts are used for the time indices for compactness. For convenience we denote

$$\begin{aligned}
 \theta(\mathbf{Q}) = & \\
 [& c_H(Q_{H,n-\tau_H}) \ c_B(Q_{B,n-\tau_B}) \ c_P(Q_{P,n-\tau_P}) \ c_{DYW}(Q_{DYW,n}) \ c_{YMC}(Q_{YMC,n}) \ c_{MC}(Q_{MC,n})].
 \end{aligned}$$

To estimate vector $\theta(\mathbf{Q})$, we follow the method developed in [54] which is described below,

1. We describe the evolution of a parameter by the following state space model

$$c_i(n+1) = f_i c_i(n) + \xi_i(n), \quad (2.19)$$

where $i = \{H, B, P, DYW, YMC, MC\}$, f_i can be tuned or initialized by some value and ξ_i is zero mean white noise with variance σ_i . The state evolution is a random walk (RW) for $f_i = 1$. Different models for the state evolution can also be considered e.g. generalized random walk (GRW), integrated random walk (IRW) etc. [54]. We enforce the integrator (in Eq. 2.18) by considering the output $z_{LM}(n) = y_{LM}(n) - y_{LM}(n-1)$. Imposing a model structure is not aligned with the DBM modelling philosophy, however, we have done it here to be consistent with other considered identification methods. We define a matrix ' H ' that carries the regression variables as

$$H(n) = [Q_{H,n-\tau_H} \quad Q_{B,n-\tau_B} \quad Q_{P,n-\tau_P} \quad Q_{DYW,n} \quad Q_{YMC,n} \quad Q_{MC,n}].$$

We can hence form a state space system with the SDPs as states,

$$c(n+1) = Fc(n) + \xi(n), \quad (2.20)$$

$$z_{LM}(n) = H(n)c(n), \quad (2.21)$$

where F is a diagonal matrix with dimensions equal to the number of parameters.

2. The SDPs are estimated sequentially by applying a Kalman Filter (KF) to Eqs. 2.20 & 2.21, while working through the available data in temporal order. The estimates obtained from the KF are also improved, using a recursive Fixed Interval Smoother (FIS) while working through the available data in reverse temporal order [54].

Since the states are SDPs, the changes in the states are expected to be frequent and abrupt over time. It is unlikely that a RW model captures the temporal variations in a parameter well. However, the idea presented in [54] can be applied i.e. the identification dataset can be sorted in an ascending order rather than in temporal order, and then estimation can be performed. With this approach the SDP variations

gets smoother, and it is more likely that a RW process can be utilized to describe the state evolution. However, the data sorting prior to estimation of the states has to be common for all variables, and to select one reference variable rarely produces satisfactory results. To cope with this, we calculate ‘partial residual series’ and estimate the parameters one by one [54], and for each input the effect of all the other inputs are subtracted from the output [54]. The FIS estimates obtained above are used as an initial guess, e.g. $\hat{c}_H^1(n)$ is a time varying estimate of the SDP associated with the flow at Heywoods.

Let $k = 1, 2, \dots$, and $i = \{H, B, P, DYW, YMC, MC\}$. Repeat the following procedure until the time series $\hat{c}_i^k(n)$ converges,

- (a) Form the partial residual series $z_{LM}^{i,k}(n) = z_{LM}(n) - \sum_{j \neq i} Q_j(n) \hat{c}_j^k(n)$, i.e. subtract the effect of all inputs, except the i^{th} input, from the output z_{LM} .
 - (b) Sort both $z_{LM}^{i,k}(n)$ and $Q_i(n)$ according to the ascending order of $Q_i(n)$. Denote the sorted variables by $\bar{z}_{LM}^{i,k}(n)$ and $\bar{Q}_i(n)$. The purpose of the sorting is to get smoother variations in the associated SDP $\hat{c}_i^k(n)$, and it is likely that a RW process can be utilized to describe it.
 - (c) Run FIS estimation using the partial residual relationship; $\bar{z}_{LM}^{i,k}(n) = \bar{Q}_i(n) \bar{c}_i^k(n)$. From the obtained estimates $\bar{c}_i^k(n)$, $\hat{c}_i^k(n)$ can be retrieved by un-sorting the estimated series in the reverse order of what was used in sorting in the previous step.
3. Finally, from the non-parametric estimates of the SDPs above, we fit suitable non-linear functions that closely describe the behaviour of the parameters with respect to the states on which they depend.

We have used the ‘sdp’ function in the CAPTAIN toolbox ([1]) to get non-parametric estimates of the SDPs in $\theta(\mathbf{Q})$. Fig. 2.10 shows the non-parametric estimates obtained from the identification dataset in Fig. 2.2. By fitting the estimates to a straight line, we

get

$$\hat{\theta}(\mathbf{Q}) = \begin{bmatrix} -9.14 \times 10^{-25} Q_H + 4.78 \times 10^{-7} \\ 8.16 \times 10^{-25} Q_B + 3.18 \times 10^{-7} \\ 3.49 \times 10^{-25} Q_P + 6.59 \times 10^{-7} \\ 2.93 \times 10^{-25} Q_{DYW} - 4.64 \times 10^{-7} \\ 8.45 \times 10^{-25} Q_{YMC} - 9.16 \times 10^{-7} \\ -1.87 \times 10^{-24} Q_{MC} - 4.38 \times 10^{-7} \end{bmatrix}^T. \quad (2.22)$$

The maximum recorded in- or out-flows in the upper part of Murray River are in the order of 10^5 ML/Day ($115,740 \text{ m}^3/\text{sec}$). The coefficients associated with each flow in the $\hat{\theta}(\mathbf{Q})$ vector are in the order of 10^{-25} to 10^{-24} . So the first term of each component in the $\hat{\theta}(\mathbf{Q})$ vector contributes in the order of 10^{-20} to 10^{-19} . This is 10^{12} to 10^{13} times smaller than the constant terms (which are in the order of 10^{-7}). So the flow dependent terms in the $\hat{\theta}(\mathbf{Q})$ vector can be safely ignored. The constant terms are in the same order of magnitude as we obtained by PEM and CT identification in Sections 2.2.1 & 2.2.3, and the signs are also in agreement with in- and out-flows. However, as mentioned earlier, the flows in the identification dataset in Fig. 2.2 do not have large flow range. We next check the SDPs obtained from identification data with larger variations.

For this, we used identification data from several years after 2000. Table 2.1 shows the variations in the data we used, the ‘straight-line fits’ of the non-parametric estimates obtained from the data, and the unexpected behaviours, if any (in blue font). We found that the non-parametric SDP estimates remain almost constant in about 95 percent of the cases, and they were similar to the ones shown in Fig. 2.10. The coefficients of the straight line fits associated with the flows are small in magnitude, and the first terms of each component in the $\hat{\theta}(\mathbf{Q})$ vectors can be ignored as explained above. Moreover, the signs and the magnitudes of the constants in the ‘straight line fits’ are in agreement with the physical structure of the river. These constants are in the same order of magnitude but their values show variations e.g. the constant part of the ‘straight line fits’ associated with the flow at Heywoods varies from 2.39×10^{-7} to 6.42×10^{-7} in Table 2.1. A bit of variation in the constant part is expected since there will often be excitation problems with some data and due to that several flows follow the same pattern.

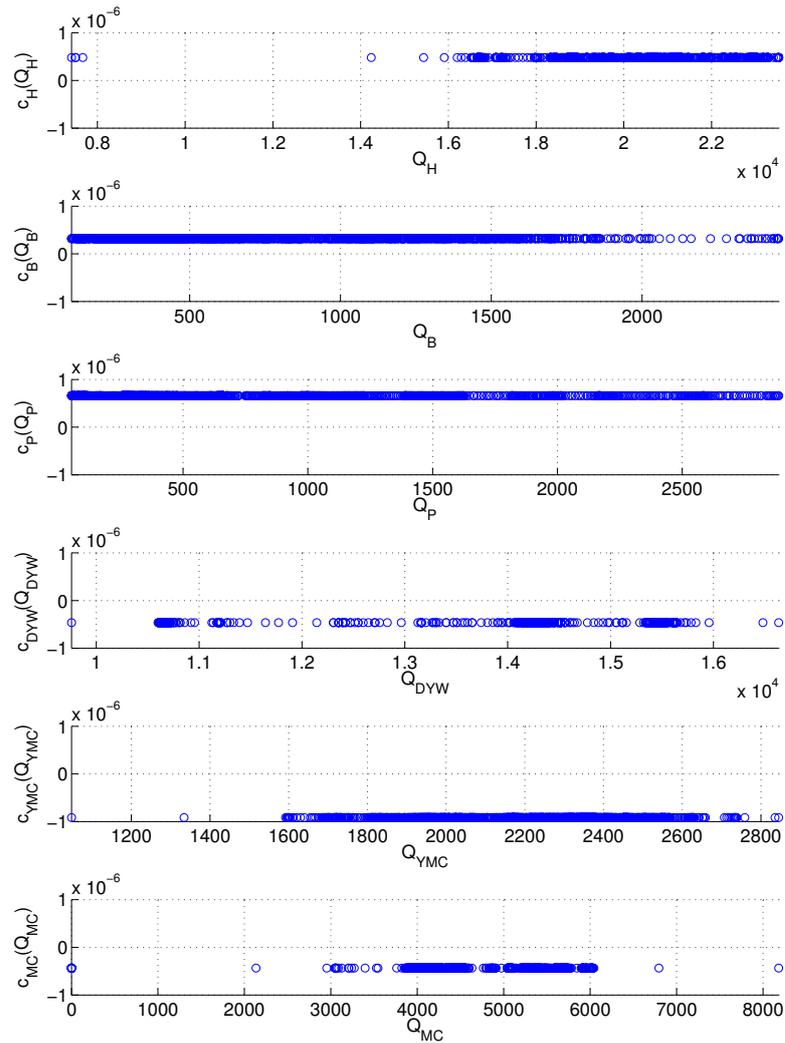


Figure 2.10: Non-parametric estimates of the state dependent parameters.

Case	Time period	Operating range (1 m ³ /sec = 86.4 ML/Day)	Straight-line fits $\hat{\theta}(Q)$
1	15 th Sep, 01 to 31 st Dec, 01	Q_H : 408 – 24,760 ML/Day, Q_B : 445 – 6,500 ML/Day, Q_P : 615 – 14,020 ML/Day, Q_{DYW} : 5,060 – 15,400 ML/Day, Q_{YMC} : 305 – 3,200 ML/Day, Q_{MC} : 2,100 – 8,900 ML/Day, γ_{LM} : 124.65 – 125.03 mAHD	$\begin{bmatrix} -1.28 \times 10^{-25} Q_H + 4.55 \times 10^{-7} \\ -2.77 \times 10^{-25} Q_B + 3.37 \times 10^{-7} \\ -8.16 \times 10^{-26} Q_P + 5.87 \times 10^{-7} \\ -8.03 \times 10^{-25} Q_{DYW} - 5.02 \times 10^{-7} \\ 3.61 \times 10^{-25} Q_{YMC} - 4.70 \times 10^{-7} \\ -2.07 \times 10^{-25} Q_{MC} - 4.10 \times 10^{-7} \end{bmatrix}^T$
2	27 th Jul, 02 to 18 th Oct, 02	Q_H : 7,920 – 25,630 ML/Day, Q_B : 681 – 3,500 ML/Day, Q_P : 969 – 6,138 ML/Day, Q_{DYW} : 5,650 – 15,200 ML/Day, Q_{YMC} : 0 – 3,050 ML/Day, Q_{MC} : 0 – 9,420 ML/Day, γ_{LM} : 122.87 – 125.05 mAHD	$\begin{bmatrix} 1.01 \times 10^{-25} Q_H + 6.42 \times 10^{-7} \\ -7.65 \times 10^{-26} Q_B + 1.41 \times 10^{-7} \\ -1.79 \times 10^{-25} Q_P + 7.93 \times 10^{-7} \\ -1.97 \times 10^{-25} Q_{DYW} - 5.33 \times 10^{-7} \\ -1.28 \times 10^{-24} Q_{YMC} - 9.02 \times 10^{-7} \\ -2.79 \times 10^{-26} Q_{MC} - 8.38 \times 10^{-7} \end{bmatrix}^T$
3	22 nd Oct, 04 to 31 st Dec, 04	Q_H : 1,865 – 18,410 ML/Day, Q_B : 430 – 3,645 ML/Day, Q_P : 1,100 – 9,330 ML/Day, Q_{DYW} : 8,520 – 13,400 ML/Day, Q_{YMC} : 150 – 2,745 ML/Day, Q_{MC} : 1,770 – 6,200 ML/Day, γ_{LM} : 124.63 – 125.04 mAHD	$\begin{bmatrix} 1.92 \times 10^{-25} Q_H + 5.96 \times 10^{-7} \\ -3.06 \times 10^{-25} Q_B + 8.17 \times 10^{-7} \\ -2.44 \times 10^{-25} Q_P + 6.43 \times 10^{-7} \\ 3.78 \times 10^{-24} Q_{DYW} - 6.59 \times 10^{-7} \\ -2.05 \times 10^{-24} Q_{YMC} - 7.42 \times 10^{-7} \\ 2.94 \times 10^{-25} Q_{MC} - 5.61 \times 10^{-7} \end{bmatrix}^T$
4	21 st Jan, 05 to 7 th Apr, 05	Q_H : 3,800 – 20,600 ML/Day, Q_B : 305 – 6,420 ML/Day, Q_P : 410 – 21,475 ML/Day, Q_{DYW} : 7,580 – 24,510 ML/Day, Q_{YMC} : 65 – 2,350 ML/Day, Q_{MC} : 1,490 – 6,700 ML/Day, γ_{LM} : 124.68 – 125.03 mAHD	$\begin{bmatrix} 1.88 \times 10^{-26} Q_H + 5.85 \times 10^{-7} \\ 5.35 \times 10^{-25} Q_B + 6.48 \times 10^{-7} \\ -5.52 \times 10^{-26} Q_P + 5.43 \times 10^{-7} \\ -3.26 \times 10^{-25} Q_{DYW} - 5.28 \times 10^{-7} \\ -1.00 \times 10^{-24} Q_{YMC} - 10.37 \times 10^{-7} \\ 6.54 \times 10^{-26} Q_{MC} - 7.46 \times 10^{-7} \end{bmatrix}^T$
5	23 th Oct, 05 to 31 st Dec, 05	Q_H : 6,120 – 22,870 ML/Day, Q_B : 645 – 16,940 ML/Day, Q_P : 1,320 – 17,355 ML/Day, Q_{DYW} : 10,780 – 31,190 ML/Day, Q_{YMC} : 15 – 2,760 ML/Day, Q_{MC} : 530 – 7,930 ML/Day, γ_{LM} : 124.71 – 124.96 mAHD	$\begin{bmatrix} -2.98 \times 10^{-25} Q_H + 5.38 \times 10^{-7} \\ -5.86 \times 10^{-28} Q_B + 7.23 \times 10^{-9} \\ -2.80 \times 10^{-13} Q_P + 9.31 \times 10^{-7} \\ 5.71 \times 10^{-26} Q_{DYW} - 5.78 \times 10^{-7} \\ -3.25 \times 10^{-23} Q_{YMC} - 6.05 \times 10^{-7} \\ -6.18 \times 10^{-7} \end{bmatrix}^T$
6	8 th Oct, 09 to 31 st Dec, 09	Q_H : 1,640 – 15,050 ML/Day, Q_B : 295 – 3,990 ML/Day, Q_P : 310 – 9,310 ML/Day, Q_{DYW} : 9,437 – 10,980 ML/Day, Q_{YMC} : 0 – 1,140 ML/Day, Q_{MC} : 465 – 2,180 ML/Day, γ_{LM} : 124.61 – 124.94 mAHD	$\begin{bmatrix} 4.84 \times 10^{-25} Q_H + 5.44 \times 10^{-7} \\ -2.60 \times 10^{-25} Q_B + 2.10 \times 10^{-7} \\ 4.86 \times 10^{-25} Q_P + 7.30 \times 10^{-7} \\ 3.81 \times 10^{-24} Q_{DYW} - 5.55 \times 10^{-7} \\ 7.09 \times 10^{-23} Q_{YMC} - 9.74 \times 10^{-7} \\ -2.09 \times 10^{-24} Q_{MC} - 6.05 \times 10^{-7} \end{bmatrix}^T$
7	1 st Jan, 11 to 7 th Apr, 11	Q_H : 680 – 25,255 ML/Day, Q_B : 1,025 – 20,834 ML/Day, Q_P : 2,788 – 29,060 ML/Day, Q_{DYW} : 15,965 – 42,685 ML/Day, Q_{YMC} : 1 – 1,525 ML/Day, Q_{MC} : 2,945 – 7,350 ML/Day, γ_{LM} : 124.61 – 124.90 mAHD	$\begin{bmatrix} 3.77 \times 10^{-26} Q_H + 2.39 \times 10^{-7} \\ 8.53 \times 10^{-26} Q_B + 2.64 \times 10^{-7} \\ 4.81 \times 10^{-7} \\ -1.33 \times 10^{-12} Q_{DYW} - 2.69 \times 10^{-7} \\ -6.96 \times 10^{-25} Q_{YMC} + 4.78 \times 10^{-7} \\ -6.45 \times 10^{-26} Q_{MC} - 3.31 \times 10^{-7} \end{bmatrix}^T$

Table 2.1: SDP estimation from data with different range of variations.

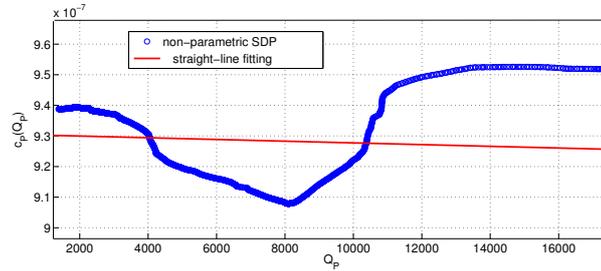


Figure 2.11: A non-parametric estimate of an SDP with straight-line fitting against a dataset from 2005.

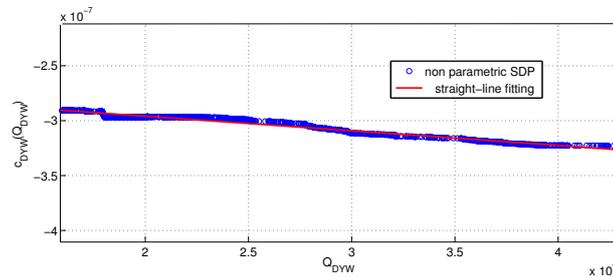


Figure 2.12: A non-parametric estimate of an SDP with straight-line fitting against a dataset from 2011.

Table 2.1 shows two cases with unexpected behaviours. Fig. 2.11 shows the estimate of the SDP associated with the inflow at Peechelba in case '5'. However, the variations are less than 5 % of the maximum value. Similarly in case '7', we found another such behaviour, and this time it corresponds to the flows over Yarrawonga Weir. Fig. 2.12 shows the corresponding non-parametric estimate. The change is almost 10 % of the maximum value. This effect can be explained as follows. Q_{YMC} is relatively constant, but it appears as an inflow instead of an outflow as the corresponding parameter is positive. The wrong sign is most likely due to the fact that Q_{YMC} is relatively smaller than the other two outflows on this particular dataset, and its effect on the water level is therefore difficult to estimate. In order to maintain the volume balance c_{DYW} must compensate for this error. Since c_{DYW} multiplies Q_{DYW} , it must take larger values during low flows and smaller values during high flows in order to compensate for an (approximately) constant flow.

In conclusion we found that there is no real need to use a non-linear model when the intended use is control. However, in models for flood prediction [17] & [93], it is benefi-

cial to incorporate non-linearities. Linear Parameter Varying (LPV) system identification (e.g. [95]) can also be employed, but the SDPs obtained by the DBM approach suggest that there is not much improvement expected from using LPV system identification.

CT model identification using SRIVC method

Here we use SRIVC method within the context of DBM modelling to identify a CT OE model of the water level in Lake Mulwala. As described earlier, an important part of the DBM modelling approach is to allow the dynamic model to be determined from the identification dataset without constraining the model parameters or imposing any specific model structure. Hence the following first order CT OE, MISO model structure was considered

$$y_{LM}(t) = \frac{b_H}{s + a_H} Q_H(t - \tau'_H) + \frac{b_B}{s + a_B} Q_B(t - \tau'_B) + \frac{b_P}{s + a_P} Q_P(t - \tau'_P) \quad (2.23)$$

$$+ \frac{b_{DYW}}{s + a_{DYW}} Q_{DYW}(t) + \frac{b_{YMC}}{s + a_{YMC}} Q_{YMC}(t) + \frac{b_{MC}}{s + a_{MC}} Q_{MC}(t).$$

The parameters a_i and b_i are to be determined by the SRIVC method, where $i = \{H, B, P, DYW, YMC, MC\}$. We expect the signs of the numerator coefficients to be in agreement with the directions of the flows and the poles to appear close to the origin. We used the same time delays as before.

We used the identification dataset in Fig. 2.2. Table 2.2 shows the identified CT transfer functions (TFs) and their discrete time (DT) versions. The estimated parameters of the CT transfer functions are in the same order of magnitude and the signs of the numerator coefficients are in agreement with the inflows and outflows. The poles are in the left half plane close to the origin. An important part of the DBM modelling that differentiates it from black-box and grey-box modelling is that the identified black-box model should always be capable of interpretation in physically meaningful terms. In this case, the most obvious explanation is that there are small unaccounted for losses in the system, but in full DBM modelling, further analysis would be necessary, possibly using additional data, in order to confirm this or investigate other possibilities.

TF corresponding to flows	CT TF	DT TF
at Heywoods	$\frac{3.21 \times 10^{-7}}{s+0.0038} e^{-70s}$	$\frac{3.20 \times 10^{-7}}{z-0.996} z^{-70}$
at Bandiana	$\frac{2.35 \times 10^{-7}}{s+0.0032} e^{-71s}$	$\frac{2.34 \times 10^{-7}}{z-0.996} z^{-71}$
at Peechelba	$\frac{5.47 \times 10^{-7}}{s+0.0049} e^{-16s}$	$\frac{5.46 \times 10^{-7}}{z-0.995} z^{-16}$
over Yarrawoonga Weir	$\frac{-2.88 \times 10^{-7}}{s+0.0036}$	$\frac{-2.87 \times 10^{-7}}{z-0.996}$
to YW main channel	$\frac{-2.12 \times 10^{-6}}{s+0.0141}$	$\frac{-2.10 \times 10^{-6}}{z-0.986}$
to Mulwala Canal	$\frac{-5.06 \times 10^{-7}}{s+0.0048}$	$\frac{-5.05 \times 10^{-7}}{z-0.995}$

Table 2.2: CT and DT transfer functions corresponding to each inflow and outflow from Lake Mulwala.

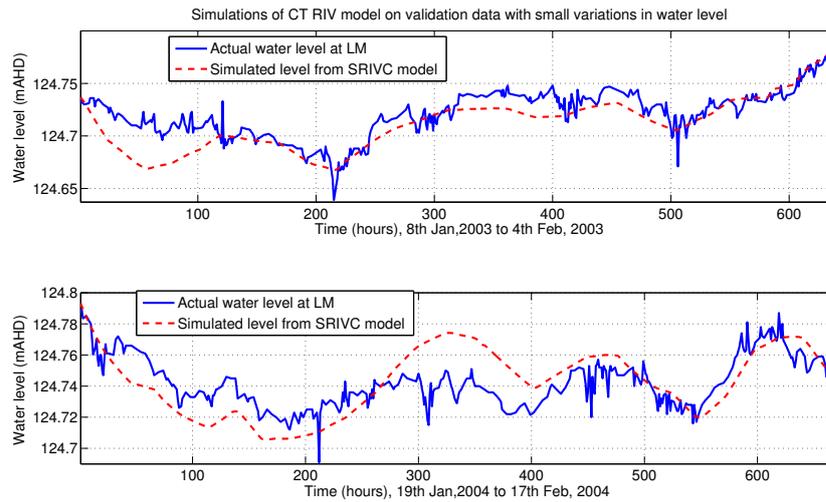


Figure 2.13: Simulation results on validation data with small variations in the water level in Lake Mulwala.

We validated the identified model against the same data as used before. Figs. 2.13 and 2.14 show the performance of the identified model. The model picks up the trends well against the data with small and large variations. We also used the CONTSID toolbox to identify higher order models but there were no or very little improvements. Some related results are added in Appendix 2.B.

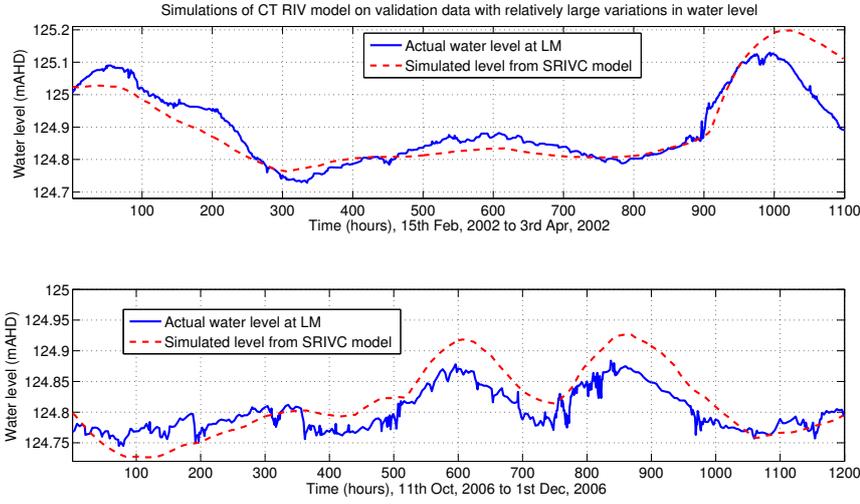


Figure 2.14: Simulation results on validation data with relatively large variations in the water level in Lake Mulwala.

2.2.5 Subspace Identification Method (SIM)

In this section we consider the SIM approach. In the literature there are a number of subspace identification techniques available (e.g. [19], [96], [97] & [87]). Here we consider Multi-Variable Output-Error State-sPace (MOESP) method [19] which uses a state space representation in innovation form

$$x(k+1) = Ax(k) + Bu(k) + Ke(k), \quad (2.24)$$

$$y(k) = Cx(k) + Du(k) + e(k), \quad (2.25)$$

where $x(k) \in \mathbb{R}^n$, $u(k) \in \mathbb{R}^m$ and $y(k) \in \mathbb{R}^p$. $e(k)$ is white noise and K is the Kalman Gain.

Multi-Variable Output-Error State-sPace method (MOESP)

In MOESP we estimate the column space of the extended observability matrix, $O_s = \begin{bmatrix} C^T & (CA)^T & \dots & (CA^{s-1})^T \end{bmatrix}^T$, where $s > n$ and n is the order of the system. The algorithm works as follows

Column space of the extended observability matrix O_s : From Eqs. 2.24 & 2.25 the

relation between the inputs and the outputs can be written as,

$$Y_{i,s,N} = O_s X_{i,N} + \mathcal{T}_s U_{i,s,N} + S_s E_{i,s,N}, \quad (2.26)$$

where

$$Y_{i,s,N} = \begin{bmatrix} y(i) & \cdots & y(i+N-1) \\ \vdots & \ddots & \vdots \\ y(i+s-1) & \cdots & y(i+s+N-2) \end{bmatrix},$$

$$U_{i,s,N} = \begin{bmatrix} u(i) & \cdots & u(i+N-1) \\ \vdots & \ddots & \vdots \\ u(i+s-1) & \cdots & u(i+s+N-2) \end{bmatrix},$$

$$E_{i,s,N} = \begin{bmatrix} e(i) & \cdots & e(i+N-1) \\ \vdots & \ddots & \vdots \\ e(i+s-1) & \cdots & e(i+s+N-2) \end{bmatrix},$$

and $X_{i,N} = [x(i) \ x(i+1) \ \cdots \ x(i+N-1)]$. Eq. 2.26 is called the “data equation”. O_s is the extended observability matrix, and \mathcal{T}_s and S_s are given by

$$\mathcal{T}_s = \begin{bmatrix} D & 0 & \cdots & 0 \\ CB & D & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{s-2}B & CA^{s-3}B & \cdots & D \end{bmatrix}, \quad S_s = \begin{bmatrix} I_p & 0 & \cdots & 0 \\ CK & I_p & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{s-2}K & CA^{s-3}K & \cdots & I_p \end{bmatrix}.$$

To estimate the column space of the matrix O_s , we remove the second and the third term on the right hand side of Eq. 2.26 by multiplying the equation with a projection matrix $\Pi_{U_{i,s,N}}^\perp$ and then by an instrumental variable matrix Z_N . For $i = s$ in Eq. 2.26 we get,

$$Y_{s,s,N} \Pi_{U_{s,s,N}}^\perp Z_N^\top = O_s X_{s,N} \Pi_{U_{s,s,N}}^\perp Z_N^\top + S_s E_{s,s,N} \Pi_{U_{s,s,N}}^\perp Z_N^\top, \quad (2.27)$$

where, $\Pi_{U_{s,s,N}}^\perp = I_N - U_{s,s,N}^\top (U_{s,s,N} U_{s,s,N}^\top)^{-1} U_{s,s,N}$ is a projection matrix which projects onto the orthogonal complement of $U_{s,s,N}$ (removes the second term of Eq. 2.26). Z_N is an instrumental variable matrix [19] comprising past input and past output data. Z_N is ideally chosen, such that $\lim_{N \rightarrow \infty} \frac{1}{N} E_{s,s,N} \Pi_{U_{s,s,N}}^\perp Z_N^\top = 0$ (removes the third term of Eq.

2.26). Depending on the choice of Z_N we have two variants of MOESP, PI-MOESP (Z_N contains past input only, $Z_N = \begin{bmatrix} U_{0,s,N} \end{bmatrix}$) and PO-MOESP (Z_N contains both past input and past output, $Z_N = \begin{bmatrix} U_{0,s,N} & Y_{0,s,N} \end{bmatrix}^\top$).

With the above mentioned properties, we take the limit $N \rightarrow \infty$ in Eq. 2.27, and get,

$$\lim_{N \rightarrow \infty} \frac{1}{N} Y_{i,s,N} \Pi_{U_{i,s,N}}^\perp Z_N^\top = \lim_{N \rightarrow \infty} \frac{1}{N} O_s X_{i,N} \Pi_{U_{i,s,N}}^\perp Z_N^\top. \quad (2.28)$$

Generally, $Z_N = \begin{bmatrix} U_{0,s,N} \\ Y_{0,s,N} \end{bmatrix}$ is always used as the choice of the instrumental variable matrix, because it provides better separation between the system dynamics and the noise [19]. Furthermore, with the following assumption,

$$\text{rank}\left(\lim_{N \rightarrow \infty} \frac{1}{N} \begin{bmatrix} X_{0,N} \\ U_{0,2s,N} \end{bmatrix} \begin{bmatrix} X_{0,N}^\top & U_{0,2s,N}^\top \end{bmatrix}\right) = n + 2sm, \quad (2.29)$$

and the application of Sylvester's Inequality [98] to Eq. 2.28, it can be shown that ([19]),

$$\text{range}\left(\lim_{N \rightarrow \infty} \frac{1}{N} Y_{s,s,N} \Pi_{U_{s,s,N}}^\perp Z_N^\top\right) = \text{range}(O_s). \quad (2.30)$$

The column space of the extended observability matrix O_s can then be estimated by taking the Singular Value Decomposition (SVD) of $G_N = \frac{1}{N} Y_{s,s,N} \Pi_{U_{s,s,N}}^\perp Z_N^\top$. The SVD gives $G_N = U_N \Sigma_N V_N^\top$, where $\text{range}(U_N) = \text{range}(G_N)$, which in the limit $N \rightarrow \infty$ is equal to $\text{range}(O_s)$. An efficient numerical implementation of the above step is given in [19].

Order and state space matrices of the system: The matrix Σ_N in the SVD contains the singular values of the system. The n largest singular values are generated by the system dynamics and the rest stem from noise. The order of the system can be estimated by finding the largest gap in the singular values. As O_s is given by $\begin{bmatrix} C^\top & (CA)^\top & \dots & (CA^{s-1})^\top \end{bmatrix}^\top$, C and A can be estimated up to a similarity transform T . Using MATLAB notation we express the estimates of C_T as,

$$\widehat{C}_T = \widehat{O}_s(1:p, 1:n).$$

An estimate of A_T can be found by solving

$$\widehat{O}_s(1 : p(s-1), 1 : n) \widehat{A}_T = \widehat{O}_s(p+1 : ps, 1 : n), \quad (2.31)$$

in a least squares sense with respect to \widehat{A}_T . After the estimates of the A and C matrices are obtained, Eqs. 2.24 & 2.25 can be used to formulate a linear regression problem from which the B and D matrices can be estimated together with the initial states $x(0)$.

Prior information in SIM

In SIM it is difficult to exploit the prior knowledge fully since we cannot fix a particular basis for the state space model during identification, especially when the data is noisy. However, we use techniques similar to those in [99] & [100] to incorporate the structure and time delays in the next subsection and in Section 2.3.4.

Validation results

We used the identification dataset in Fig. 2.2 where the delays were incorporated i.e. $\{[Q_H(k - \tau_H), Q_B(k - \tau_B), Q_P(k - \tau_P), Q_{DYW}(k), Q_{YMC}(k), Q_{MC}(k)], [y_{LM}(k)]\}_{k=\tau_{\max}}^N$. $\tau_{\max} = \max(\tau_H, \tau_B, \tau_P)$ is the maximum time delay. The time delays from Section 2.2.1 were used, i.e. $\tau_H = 70$, $\tau_B = 71$ and $\tau_P = 16$ hours. With $s = 5$ in Eq. 2.26, we got the singular values

$$\left[592.02 \quad 0.055 \quad 0.0012 \quad 1.26 \times 10^{-5} \quad 9.41 \times 10^{-8} \right].$$

We chose a first order model since the biggest gap is between the first and second singular values. The matrices of the identified state space model were,

$$A = \begin{bmatrix} 1 \end{bmatrix}, \quad C = \begin{bmatrix} -10.88 \end{bmatrix},$$

$$B = \begin{bmatrix} -5.73 \times 10^{-8} & -6.07 \times 10^{-8} & -7.66 \times 10^{-8} & 5.78 \times 10^{-8} & 9.20 \times 10^{-8} & 6.10 \times 10^{-8} \end{bmatrix}.$$

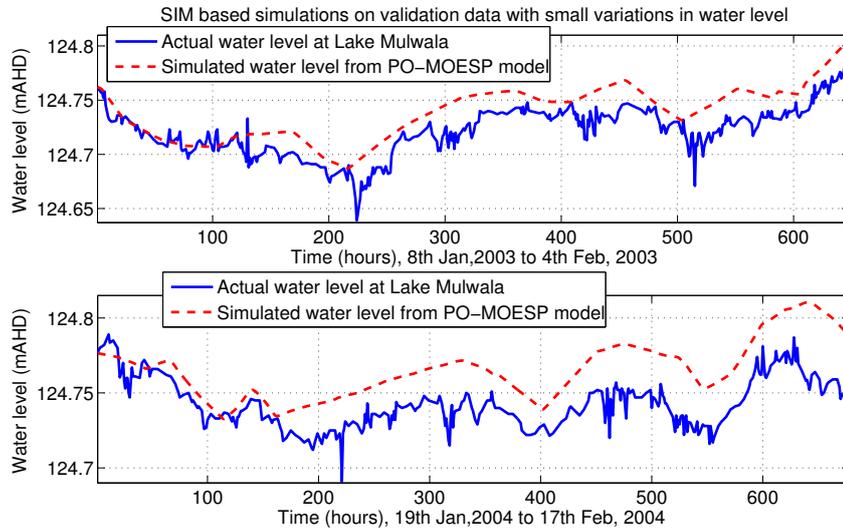


Figure 2.15: Simulation results on validation data with small variations in the water level in Lake Mulwala.

The actual value of the element in matrix A was 0.99999841, which we rounded off to 1, such that the system has an integrator. The fact that the model should contain an integrator can also be incorporated by considering $y_{LM}(n+1) - y_{LM}(n)$ as the output. The D matrix was forced to be zero as there is no direct coupling between the inputs and the output. This was achieved by simply removing the regression vectors corresponding to the unknown entries in the D matrix in the least squares problem for the estimation of B , D and the initial state. A similar approach is considered in [99] & [100] to enforce structure and prior information in subspace identification. If we compute the transfer function of the SIM based model above, we get $\hat{\theta}_{SS} = [6.23 \times 10^{-7} \quad 6.61 \times 10^{-7} \quad 8.34 \times 10^{-7} \quad -6.29 \times 10^{-7} \quad -10.01 \times 10^{-7} \quad -6.64 \times 10^{-7}]$ as the parameter vector for the predictor model in Eq. 2.6. $\hat{\theta}_{SS}$ is close to the values obtained with the other methods.

The validation results are shown in Figs. 2.15 & 2.16. The model picks up the trends in the data well. In Fig. 2.15 the gap between the actual and the simulated water level is increasing. Again, the reason is that the model contains an integrator and the errors are accumulated over time.

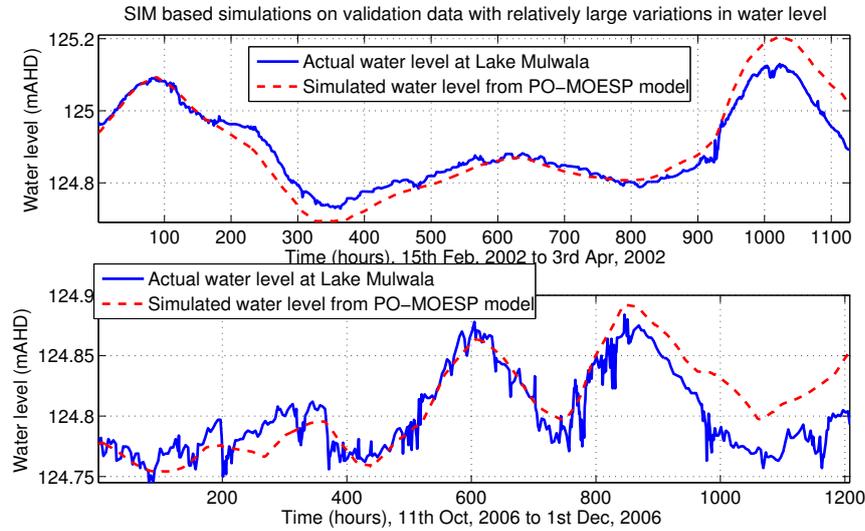


Figure 2.16: Simulation results on validation data with relatively large variations in the water level in Lake Mulwala.

2.2.6 Comparison of results

In Figs. 2.17 & 2.18 the simulation results of the identified models from Sections 2.2.1, 2.2.3, 2.2.4 & 2.2.5 are plotted together. The identified models pick up the trends in the validation data well. The simulation results of PEM and CT OE models are almost the same and that is why y_{PEM} is hidden under $y_{CT\ OE}$ on some figures. The Sum of the Squared Prediction Errors (SSPE) on validation data are shown in Table 2.3. The SSPE is given by

$$SSPE = \frac{1}{N} \sum_{t=1}^N (y_{LM}(t) - \hat{y}_{LM}(t, \theta))^2. \quad (2.32)$$

It is hard to nominate any technique as the best for the MISO case, however, PEM and CT OE models performed consistently well on the validation data.

All methods could incorporate and capture the available prior information, and the signs of the identified model parameters were in agreement with the flow directions. In DBM we did not a-priori incorporate the prior information since the DBM modelling philosophy is to allow the dynamic model structure to be identified directly from the available data, and constraining the data and imposing a particular model structure are usually not in agreement with the DBM modelling philosophy. No major computational

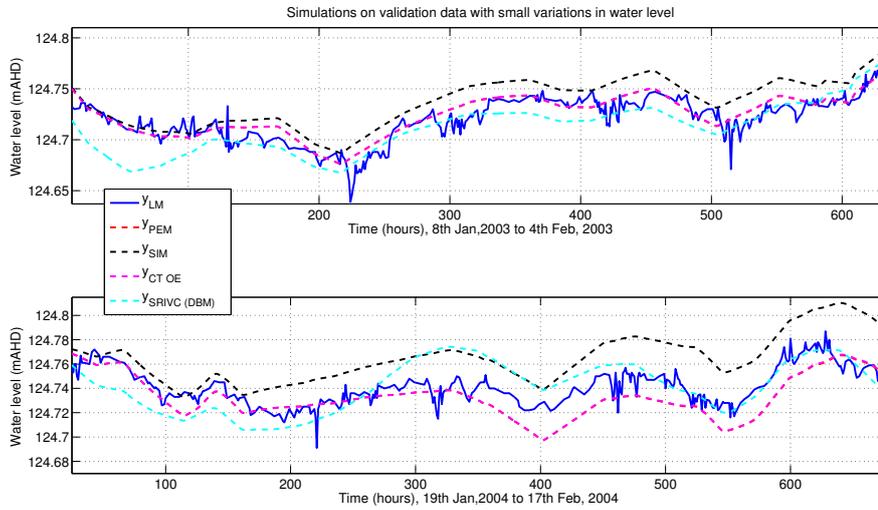


Figure 2.17: Simulation results on validation data with small variations in the water level in Lake Mulwala.

Dataset	PEM	CT OE	SRIVC (DBM)	SIM
2002	1.50×10^{-3}	1.90×10^{-3}	3.50×10^{-3}	2.10×10^{-3}
2003	0.97×10^{-4}	0.98×10^{-4}	2.60×10^{-4}	4.02×10^{-4}
2004	1.80×10^{-4}	1.75×10^{-4}	2.71×10^{-4}	6.85×10^{-4}
2006	3.81×10^{-4}	3.27×10^{-4}	1.10×10^{-3}	4.98×10^{-4}

Table 2.3: SSPE for the water level simulations in Lake Mulwala.

issues were observed while solving optimisation problems in PEM or CT methods because the optimisation problems were small scale and convex in the parameters. SIM is reliable in terms of computational complexity as it is a non-iterative technique which is not based on solving an optimisation problem.

The obtained models can be used in model based control techniques such as Model Predictive Control (MPC), and we will use the PEM based model in the proceeding chapters to control water level in Lake Mulwala.

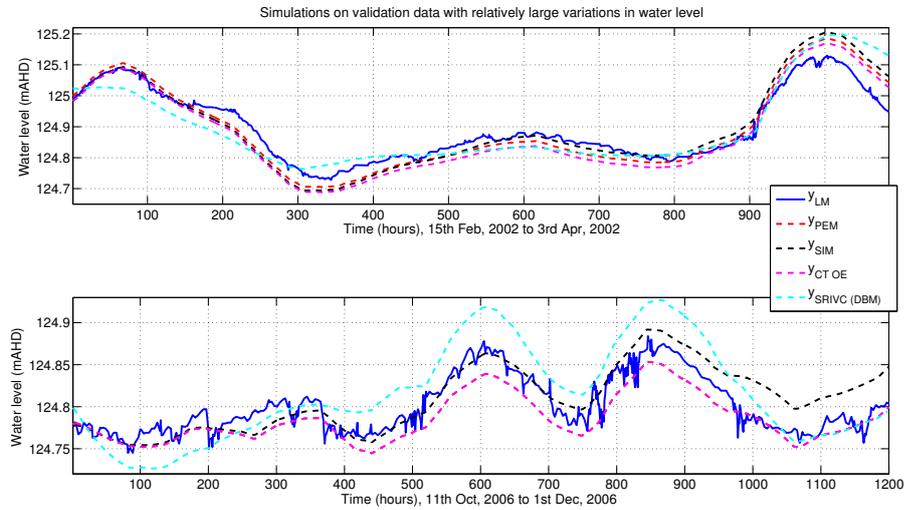


Figure 2.18: Simulation results on validation data with relatively large variations in the water level in Lake Mulwala.

2.3 Modelling water level in Lake Mulwala and flow at Doctors Point

The water level at Doctors Point is another important variable in the upper part of Murray River. The rate of fall of the water level at Doctors Point has to be kept below 15 cm/Day to avoid river bank slumping. In this section we consider the identification methods described in Section 2.2 to identify a MIMO model with the water level in Lake Mulwala y_{LM} , and the flow at Doctors Point Q_{DP} , as output variables. The water level at Doctors Point is then obtained from the flow using the rating curve shown in Fig. 2.19. We implemented our own identification routines for all methods except for the continuous time state space model, where we used CONTSID toolbox [2] in Matlab. In Section 2.4 we compare the models obtained in this section and discuss their simulation performance.

2.3.1 PEM approach

In PEM we prefer to identify models for each output separately. It becomes more difficult when the models have common parameters, but in this case the models can be

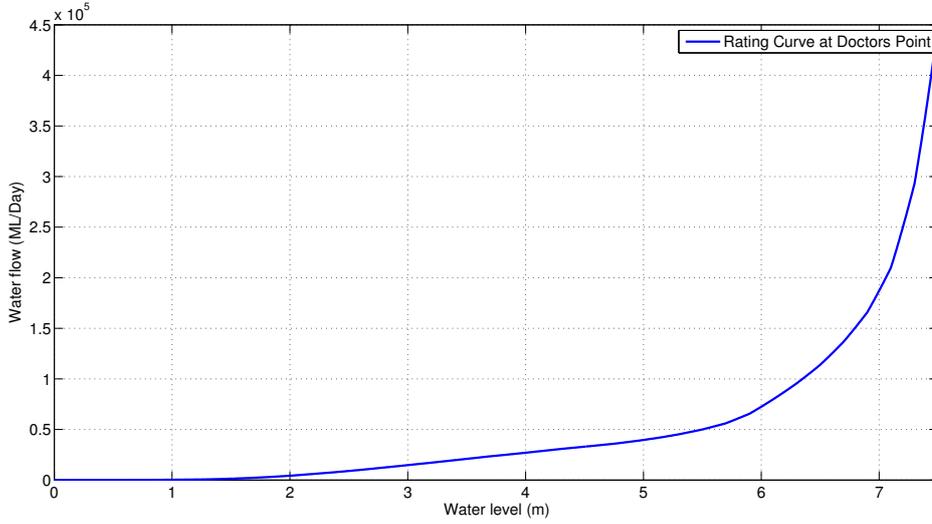


Figure 2.19: Rating curve between the flow and water level at Doctors Point.

parametrised independently. For the water level in Lake Mulwala we use the same model as before. From Fig. 2.1 the flow at Doctors Point can be described by the following OE model,

$$\hat{Q}_{DP}(n|\theta) = \alpha \hat{Q}_{DP}(n-1|\theta) + \beta_H Q_H(n - \tau'_H) + \beta_B Q_B(n - \tau'_B), \quad (2.33)$$

where Q_H and Q_B are the flows at Heywoods and Bandiana, and β_H and β_B are the parameters associated with them. τ'_H and τ'_B are the time delays from Heywoods and Bandiana to Doctors Point respectively. $\theta = [\alpha \ \beta_H \ \beta_B]^\top$ is the unknown parameter vector.

The cross-correlation method was used to estimate the time delays τ'_H and τ'_B . For this case we estimated the time delay from Doctors Point to Corowa ($\tau_{D \rightarrow C}$) (see Fig. 2.1), and obtained estimates of τ'_H and τ'_B by subtracting $\hat{\tau}_{D \rightarrow C}$ from the time delay estimates from Heywoods to Corowa ($\hat{\tau}_{H \rightarrow C} = 51$ hours) and from Bandiana to Corowa ($\hat{\tau}_{B \rightarrow C} = 52$ hours) found in Appendix 2.1.5. Using this approach we got $\hat{\tau}'_H = 1$ hour and $\hat{\tau}'_B = 2$ hours. The parameters in the θ vector are estimated by minimising

$$\hat{\theta} = \arg \min_{\theta} \frac{1}{N - \tau'_{\max}} \sum_{n=\tau'_{\max}+1}^N (Q_{DP}(n) - \hat{Q}_{DP}(n|\theta))^2, \quad (2.34)$$

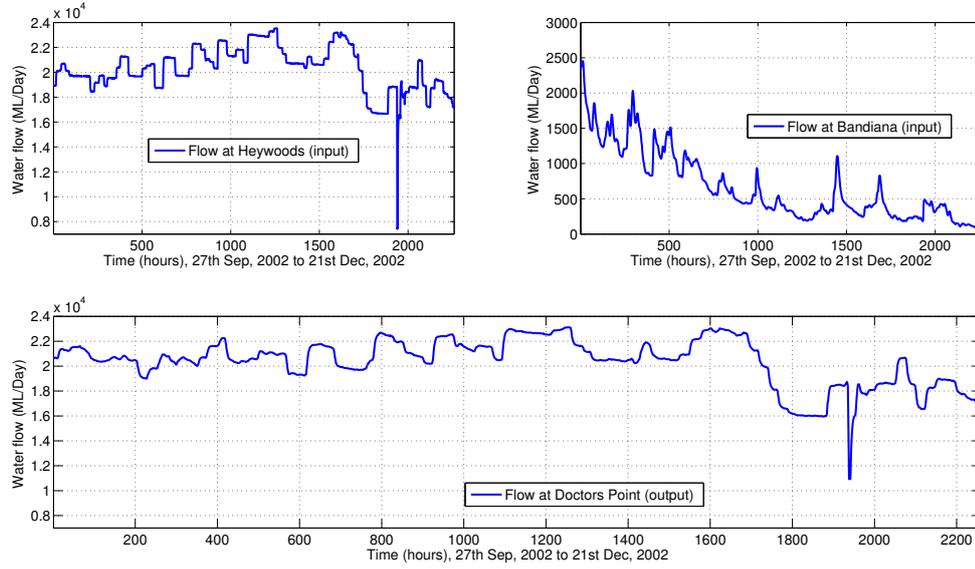


Figure 2.20: Identification dataset.

where $\tau'_{\max} = \max(\hat{\tau}'_H, \hat{\tau}'_B)$. Using the identification dataset in Fig. 2.20, the estimates $\hat{\alpha} = 0.318$, $\hat{\beta}_H = 0.663$ and $\hat{\beta}_B = 0.679$ were obtained. This seems reasonable as the model represents lowpass filters with DC gains close to 1. Validation results are shown in Fig. 2.21. The rating curve in Fig. 2.19 was used to calculate the water level at Doctors Point. The results for the same data for the water level in Lake Mulwala can be found in the upper and lower graphs of Figs. 2.6 & 2.7 respectively.

2.3.2 ML approach

In this section we extend the Maximum Likelihood (ML) approach from Section 2.2.2 to the MIMO case. We consider noise free models of water level in Lake Mulwala x_{LM} and flow at Doctors Point x_{DP} ,

$$x_{LM}(n+1|\theta) = x_{LM}(n|\theta) + c_H Q_H(n - \tau_H) + c_B Q_B(n - \tau_B) + c_P Q_P(n - \tau_P) + c_{DYW} Q_{DYW}(n) + c_{YMC} Q_{YMC}(n) + c_{MC} Q_{MC}(n), \quad (2.35)$$

$$x_{DP}(n|\theta) = \alpha x_{DP}(n-1|\theta) + \beta_H Q_H(n - \tau'_H) + \beta_B Q_B(n - \tau'_B), \quad (2.36)$$

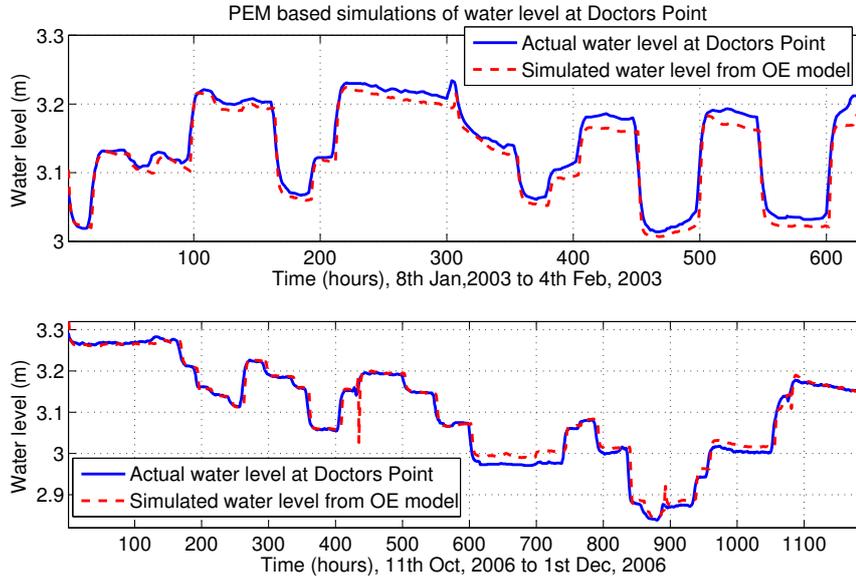


Figure 2.21: Simulation results on validation data.

where the unknown parameter vector is

$$\theta = \left[c_H \quad c_B \quad c_P \quad c_{DYW} \quad c_{YMC} \quad c_{MC} \quad \alpha \quad \beta_H \quad \beta_B \right],$$

and time delay estimates obtained in Sections 2.2.1 & 2.3.1 are considered as prior knowledge. We further assume that the measured water level in the lake y_{LM} and the measured flow at Doctors Point Q_{DP} are noisy versions of x_{LM} and x_{DP} respectively i.e.

$$y_{LM}(n) = x_{LM}(n|\theta) + e_{LM}(n),$$

$$Q_{DP}(n) = x_{DP}(n|\theta) + e_{DP}(n),$$

where $\{e_{LM}(n)\}$ and $\{e_{DP}(n)\}$ are sequences of independent and identically distributed (i.i.d), normal and zero-mean random variables with covariance matrix Λ . The prediction errors are given by

$$\varepsilon(n|\theta) = \begin{bmatrix} \varepsilon_{LM}(n|\theta) \\ \varepsilon_{DP}(n|\theta) \end{bmatrix} = \begin{bmatrix} y_{LM}(n) - x_{LM}(n|\theta) \\ Q_{DP}(n) - x_{DP}(n|\theta) \end{bmatrix}, \quad \text{for } n = 1, 2, \dots, N, \quad (2.37)$$

where N is the number of identification data points. Using the above assumptions the joint density function of the prediction errors at time n is given by

$$f_{\varepsilon,n}(\varepsilon(n)|\theta) \sim \mathcal{N}([0 \ 0]^\top, \Lambda) = \frac{1}{\sqrt{(2\pi)^2|\Lambda|}} e^{(-\frac{1}{2}(\varepsilon(n|\theta)^\top \Lambda^{-1} \varepsilon(n|\theta))}, \quad (2.38)$$

where $|\Lambda|$ is the determinant of Λ . The negative log-likelihood function is (see [18]),

$$-\log \mathcal{L} = \text{constant} + \frac{N}{2} \log|\Lambda| + \frac{1}{2} \sum_{n=1}^N \varepsilon^\top(n|\theta) \Lambda^{-1} \varepsilon(n|\theta). \quad (2.39)$$

Apart from the parameters in θ , the elements in the covariance matrix Λ are also regarded as unknown parameters, and the estimates are given by

$$(\hat{\theta}, \hat{\Lambda}) = \arg \min_{\theta, \Lambda} \frac{1}{N - \tau_{\max}} \sum_{n=\tau_{\max}+1}^N \varepsilon^\top(n|\theta) \Lambda^{-1} \varepsilon(n|\theta) + \frac{N - \tau_{\max}}{2} \log|\Lambda|, \quad (2.40)$$

where $\tau_{\max} = \max(\tau_H, \tau_B, \tau_P, \tau'_H, \tau'_B) = 71$. If Λ is unknown and not parametrised through θ , then it is possible to minimise Eq. 2.40 analytically with respect to Λ for every fixed θ , and the solution is given by [18]

$$\hat{\Lambda}(\theta) = \frac{1}{N - 71} \sum_{n=72}^N \varepsilon(n|\theta) \varepsilon^\top(n|\theta). \quad (2.41)$$

A numerical scheme which iteratively estimates the error covariance matrix (using Eq. 2.41) and the parameters in the θ vector until they converge works as follows

Let $\hat{\Lambda}_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. Compute for $i = 1, 2, \dots$, till $\|\hat{\theta}_i - \hat{\theta}_{i-1}\| < \epsilon$, where ϵ is a small number.

1. Parameters estimate,

$$\hat{\theta}_i = \arg \min_{\theta} \frac{1}{N - 71} \sum_{n=72}^N \varepsilon^\top(n|\theta) \hat{\Lambda}_{i-1}^{-1} \varepsilon(n|\theta).$$

2. Covariance matrix estimate,

$$\hat{\Lambda}_i = \frac{1}{N - 71} \sum_{n=72}^N \varepsilon(n|\hat{\theta}_i) \varepsilon^\top(n|\hat{\theta}_i).$$

The dataset in Fig. 2.2 along with the flow data at Doctors Point shown in Fig. 2.20 was used for identification. In the output dataset there is an order of magnitude difference in the numerical values for the water level in Lake Mulwala and the flow at Doctors Point. To avoid numerical problems, we scaled the flows and water levels as below,

- All flows in the dataset were divided by the maximum flow value of the dataset. This helped in scaling the flows between 0 and 1.
- The mean of the water level in the dataset was subtracted from the water levels, which spread the data values around zero.

We used the above mentioned iterative scheme, and the scheme converged quickly, i.e. with in a few iterations (3 to 5), there was no significant change in the estimated parameters $\hat{\theta}_i$. The estimate of the parameters corresponding to unscaled data was

$$[6.25 \times 10^{-7} \quad 9.66 \times 10^{-7} \quad 5.92 \times 10^{-7} \quad -6.20 \times 10^{-7} \quad -9.80 \times 10^{-7} \quad -6.69 \times 10^{-7} \\ 0.33 \quad 0.65 \quad 0.69].$$

The inverse error covariance matrix was

$$\hat{\Lambda}^{-1} = \begin{bmatrix} 2.95 \times 10^{-4} & 2.90 \times 10^{-6} \\ 2.90 \times 10^{-6} & 3.58 \times 10^{-4} \end{bmatrix}.$$

The first six parameters of the parameter vector $\hat{\theta}_{ML}$ belong to the predictor in Eq. 2.6 (based on Eq. 2.35) and the last three parameters belong to the predictor in Eq. 2.33 (based on Eq. 2.36). The signs of the parameters support the physical structure of the river, and they are in agreement with the results obtained by PEM. Moreover, the covariance matrix is nearly diagonal, which shows that the ML estimate of θ should be nearly the same as the PEM estimate, and from Sections 2.2.1 & 2.3.1, we can see that the parameter estimates are numerically close. Figs. 2.22 & 2.23 show the performance of the identified model. The models pick up the trends in the data well.

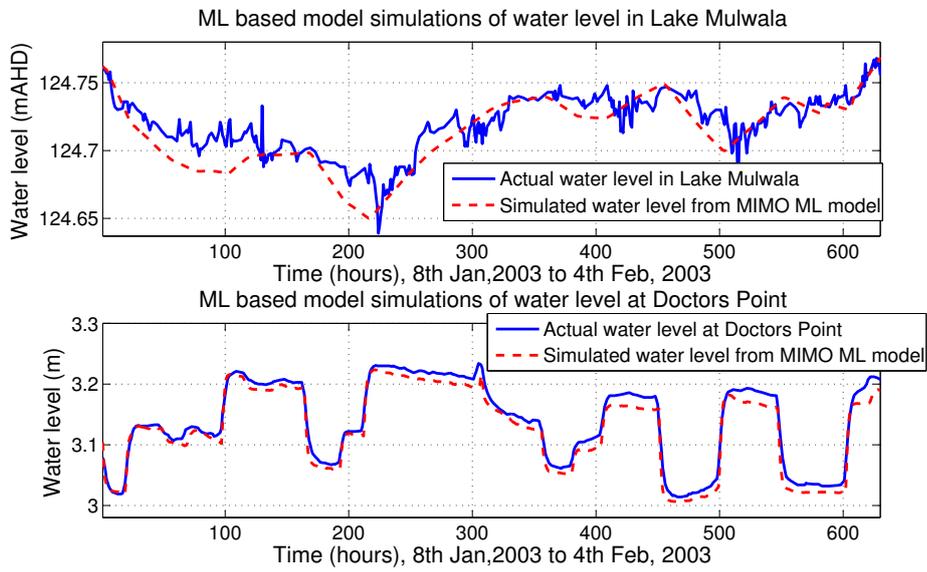


Figure 2.22: Simulation results on validation data.

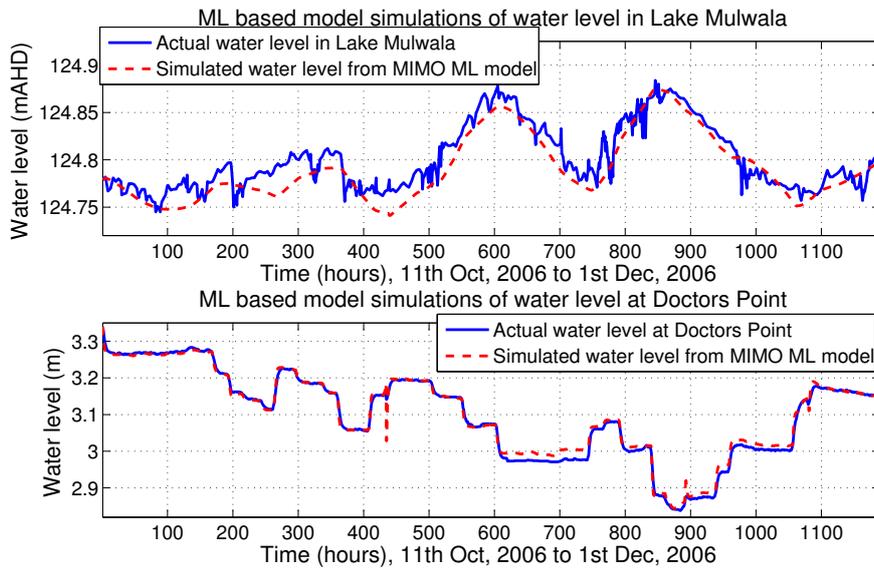


Figure 2.23: Simulation results on validation data.

2.3.3 Continuous Time (CT) identification

In this section we use CT identification techniques to find a MIMO model. There are a few techniques developed in CT system identification which can identify MIMO state space models. Here we used the canonical state space (SS) model identification via Poisson Moment Functional (PMF) which is implemented in the CONTSID toolbox [2]. Later in this section, we also identify a separate CT MISO model for the flow at Doctors Point.

The details of the SS-PMF are available in [101]. The method consists of two steps

1. The differential equation representing a dynamical system is first converted into simple algebraic equations through PMF transforms.
2. The unknown parameters are then estimated by the standard least squares (LS), or instrumental variable (IV) methods. The unknown parameters are the elements in the matrices of a CT state space model.

The function `sslsqpmf` in CONTSID toolbox [2] finds a controllable canonical state space model from the given data and uses the LS approach to find the unknown parameters. We used the same identification dataset as before and scaled the flows and water levels as in the previous section. The time delays were taken as prior knowledge. We found a second order CT state space model with the following matrices. The subscript C represents continuous time,

$$A_C = \begin{bmatrix} -1.07 \times 10^{-7} & 2.23 \times 10^{-6} \\ -1.65 \times 10^{-5} & -2.69 \times 10^{-4} \end{bmatrix}, \quad C_C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

$$B_C = \begin{bmatrix} 8.12 \times 10^{-7} & -6.98 \times 10^{-8} & 3.99 \times 10^{-6} & -3.12 \times 10^{-6} & -4.76 \times 10^{-6} & -2.86 \times 10^{-6} \\ 2.76 \times 10^{-4} & 3.29 \times 10^{-4} & 1.37 \times 10^{-5} & 1.40 \times 10^{-6} & -8.55 \times 10^{-5} & -3.67 \times 10^{-5} \end{bmatrix}.$$

The identified matrices are not in agreement with physical reality. E.g. the second output i.e. the flow at Doctors Point cannot depend on the flows at the downstream end, and hence the last four entries of the second row in matrix B_C should be zero. Similarly, in matrix A_C the element at location (2,1) should be zero as well. In principle we could reformulate the LS problem formulation and force certain entries in the A_C and B_C matrices to

zero. However, this was not possible as we did not have access to the internal variables of the functions in the CONTSID toolbox. Nonetheless, the process is straightforward and is described in Section 2.3.4 where we use SIM to identify a MIMO model.

We next find a separate CT MISO model for the flow at Doctors Point in order to avoid models which are not in agreement with physical reality. We consider the following first order differential equation to derive a predictor for the flow at Doctors Point,

$$\frac{dQ_{DP}(t)}{dt} + \alpha' Q_{DP}(t) = \beta'_H Q_H(t-1) + \beta'_B Q_B(t-2), \quad (2.42)$$

The unknown parameter vector is $\theta = \left[\alpha' \quad \beta'_H \quad \beta'_B \right]$, and the solution of Eq. 2.42 is

$$Q_{DP}(t) = e^{\alpha'(t_0-t)} Q_{DP}(t_0) + \beta'_H \left(Q_H(t_0-1) \frac{(1 - e^{\alpha'(t_0-t)})}{\alpha'} \right) + \beta'_B \left(\left(\frac{Q_B(t_0-2)}{\alpha'} - \frac{Q_B(t-2) - Q_B(t_0-2)}{(t-t_0)\alpha'^2} \right) (1 - e^{\alpha'(t_0-t)}) + \frac{Q_B(t-2) - Q_B(t_0-2)}{\alpha'} \right).$$

where we have assumed piecewise constant flows at Heywoods (zoh) and piecewise linear flows at Bandiana (foh). With a sampling interval of $T = 1$ hour, we get the following OE-type predictor ($t = t_0 + 1$)

$$\hat{Q}_{DP}(t|\theta) = e^{-\alpha'} \hat{Q}_{DP}(t-1|\theta) + \beta'_H \left(Q_H(t-2) \frac{(1 - e^{-\alpha'})}{\alpha'} \right) + \beta'_B \left(\left(\frac{Q_B(t-3)}{\alpha'} - \frac{Q_B(t-2) - Q_B(t-3)}{\alpha'^2} \right) (1 - e^{-\alpha'}) + \frac{Q_B(t-2) - Q_B(t-3)}{\alpha'} \right). \quad (2.43)$$

The parameters in the θ vector are then estimated by minimising the sum of the squared prediction errors i.e.

$$\hat{\theta} = \arg \min_{\theta} \frac{1}{N-2} \sum_{t=3}^N (Q_{DP}(t) - \hat{Q}_{DP}(t|\theta))^2. \quad (2.44)$$

We got the estimate of the parameter vector as $\hat{\theta} = \left[1 \quad 0.972 \quad 0.998 \right]$. Table 2.4 shows the identified CT transfer functions (TFs) corresponding to the inflows at Heywoods and Bandiana. The table also shows the DT approximation of the CT-TFs with zoh-

TF corresponding to inflows	CT-TF	DT-TF (zoh)	DT-TF (foh)
at Heywoods	$\frac{0.972}{s+1}$	$\frac{0.615}{z-0.368}$	—
at Bandiana	$\frac{0.998}{s+1}$	$\frac{0.631}{z-0.368}$	$\frac{0.367z+0.264}{z-0.368}$

Table 2.4: CT and DT transfer functions corresponding to the inflows at Heywoods and Bandiana.

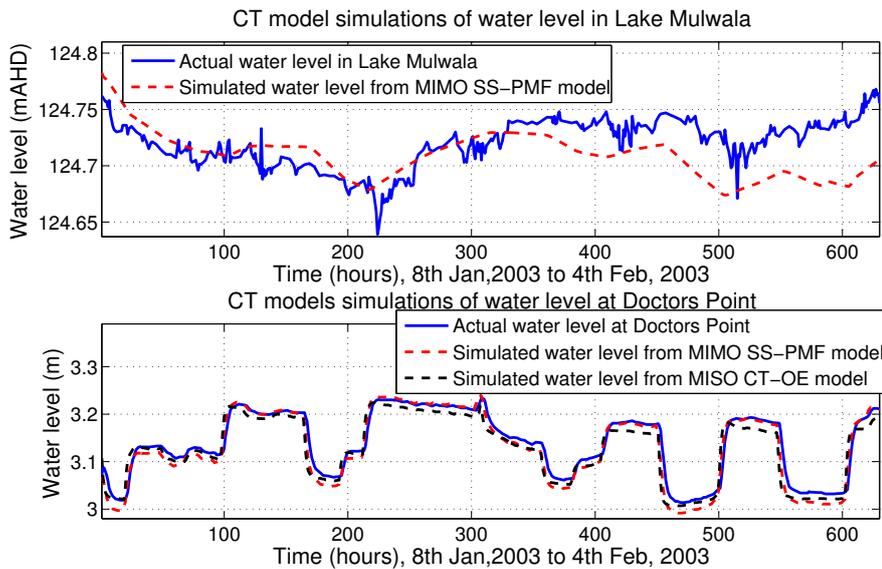


Figure 2.24: Simulation results on validation data.

approximation applied to both the transfer functions, and foh-approximation applied to the TF corresponding to the flows at Bandiana. The identified TFs are reasonable as they represent lowpass filters with DC gains close to 1. Moreover, as per expectations the parameters in the zoh approximation of the CT-TFs are similar to the parameter estimates obtained by PEM and ML in Sections 2.3.1 & 2.3.2 respectively.

We used the identified SS-PMF based MIMO model and the CT-OE model (Eq. 2.43) on different validation data. Figs. 2.24 & 2.25 show the simulation results, and in general the models pick up the trends in the data well. Although the SS-PMF method did not give parameters in agreement with physical reality and the model did not perform very well on the dataset from 2003, however, it still gave a good simulation performance on the dataset from 2006, see Fig. 2.25.

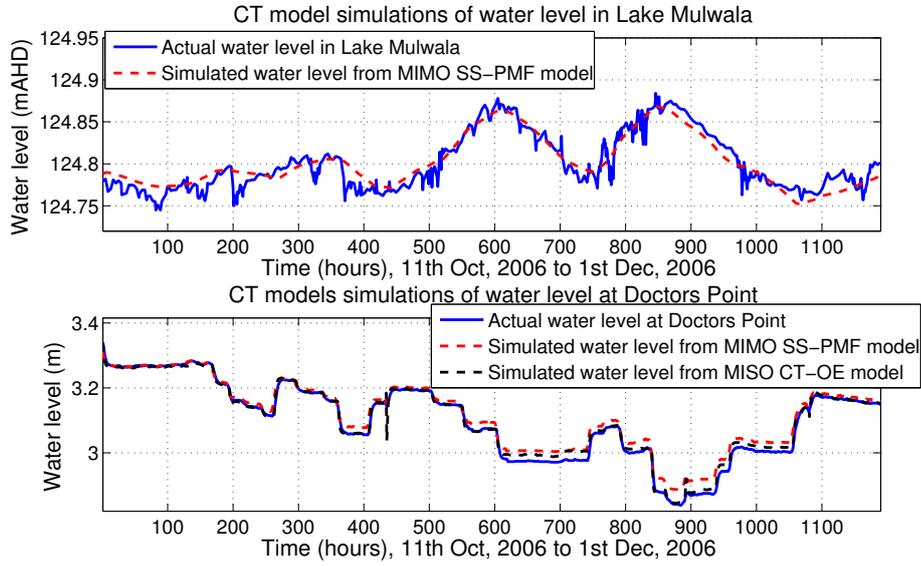


Figure 2.25: Simulation results on validation data.

2.3.4 SIM approach

The identification dataset in Fig. 2.2 was used along with the flow at Doctors Point shown in Fig. 2.20. The inputs and outputs were taken as $\{[Q_H(k-70), Q_B(k-71), Q_P(k-16), Q_{DYW}(k), Q_{YMC}(k), Q_{MC}(k)], [y_{LM}(k), Q_{DP}(k-69)]\}_{k=72}^N$. To avoid numerical problems, all flows and water levels were scaled as in Section 2.3.2. In the PO-MOESP algorithm (Section 2.2.5) we used $s = 5$, and we found the first five singular values [7.672 3.055 0.371 0.081 0.044]. There are no large gaps between the singular values, but the relative gap between the second and the third is the largest, and we chose a second order model. As expected, the identified matrices A , B and C had no zero entries, while the D matrix was forced to be zero in the same way as we did in Section 2.2.5. We obtained the following matrices

$$A = \begin{bmatrix} 1.0008 & 0.0125 \\ -0.0472 & 0.9103 \end{bmatrix}, \quad C = \begin{bmatrix} -0.7862 & 0.0063 \\ 0.0142 & -0.3398 \end{bmatrix},$$

$$B = \begin{bmatrix} 0.017 & 0.02 & -0.025 & 0.019 & 0.031 & 0.016 \\ -0.23 & -0.13 & -0.076 & -0.044 & -0.025 & 0.030 \end{bmatrix}.$$

The system matrices above and in the rest of this section correspond to the scaled dataset.

The model has no or little physical explanation of the states. As system matrices are only given up to a similarity transform, it is very rare that they describe the physical structure of the river. We noticed this behaviour in Section 2.3.3 as well. As there are only two states, it is relatively easy to find a transform which incorporates most of the prior knowledge but as the number of states increases, it gets harder. In order to give physical meaning to the states using the prior information we followed two strategies.

Strategy 1

After estimating the A and C matrices, we chose a transformation matrix T which gave a diagonal matrix C_T . The reason for finding a diagonal C was to decouple the states such that each state represents an output. Many such transformation matrices exist, and we picked one which diagonalized the C matrix and gave a small value in element (2,1) of the A_T matrix. The transformation matrix was

$$T = \begin{bmatrix} 0.1199 & 0.0500 \\ 0.0050 & 6.2331 \end{bmatrix}.$$

After adjusting A_T and C_T we formulated the linear regression problem (which provides B_T and D_T) such that the downstream flows did not affect the flow at Doctors Point. To achieve this, certain elements of the B_T matrix was forced to zero in the same way as the D_T matrix was forced to zero above. After the adjustments we got the following system matrices

$$A_T = \begin{bmatrix} 1.002 & 0.6879 \\ -0.00098 & 0.9094 \end{bmatrix}, \quad C_T = \begin{bmatrix} -0.0943 & 0 \\ 0 & -2.117 \end{bmatrix},$$

$$B_T = \begin{bmatrix} 0.189 & 0.299 & -0.277 & 0.113 & 0.229 & 0.161 \\ -0.042 & -0.038 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The first entry in the first row of the matrix A_T indicates the existence of an integrator and the first entry in the second row is small so that the water level in Lake Mulwala hardly influences the flow at Doctors Point. The eigenvalues of the matrix A_T are 0.993

and 0.917 which shows that the model is stable. All the entries in the second row of the matrix B_T which correspond to the downstream flows are zero as they cannot affect the flow at Doctors Point. The corresponding transfer functions are no longer integrator-delay models, but they are “close”. E.g. the transfer function model between the flow at Peechelba and the water level in Lake Mulwala is

$$\begin{aligned} \frac{y_{LM}(z)}{Q_P(z)} &= \frac{0.026z - 0.024}{z^2 - 1.911z + 0.912} z^{-16} \\ &= \frac{0.026(z - 0.923)}{(z - 0.987)(z - 0.924)} z^{-16} \approx \frac{0.026}{(z - 0.987)} z^{-16}. \end{aligned}$$

There is a near pole-zero cancellation and a pole close to one. Next we propose another strategy which enforces an integrator in the model.

Strategy 2

After estimating the A and C matrices, we chose a transformation matrix T' such that $CT' = I_2$ which decouples the states. The transformation matrix was

$$T' = \begin{bmatrix} -1.272 & -0.024 \\ -0.053 & -2.944 \end{bmatrix},$$

and then we applied the transformation to the extended observability matrix (in the PO-MOESP algorithm) so that the new A matrix is $A_{T'} = T'^{-1}AT'$. We further modified the least squares problem in Eq. 2.31 such that the first column in the $A_{T'}$ matrix was $[1 \ 0]^T$, reflecting that the flow at Doctors Point does not depend on the water level in Lake Mulwala, and enforcing an integrator. The linear regression problem (which provides $B_{T'}$ and $D_{T'}$) was then formulated as above such that the downstream flows did not affect the flow at Doctors Point. We got the following system matrices

$$A_{T'} = \begin{bmatrix} 1 & 0.011 \\ 0 & 0.911 \end{bmatrix}, \quad C_{T'} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

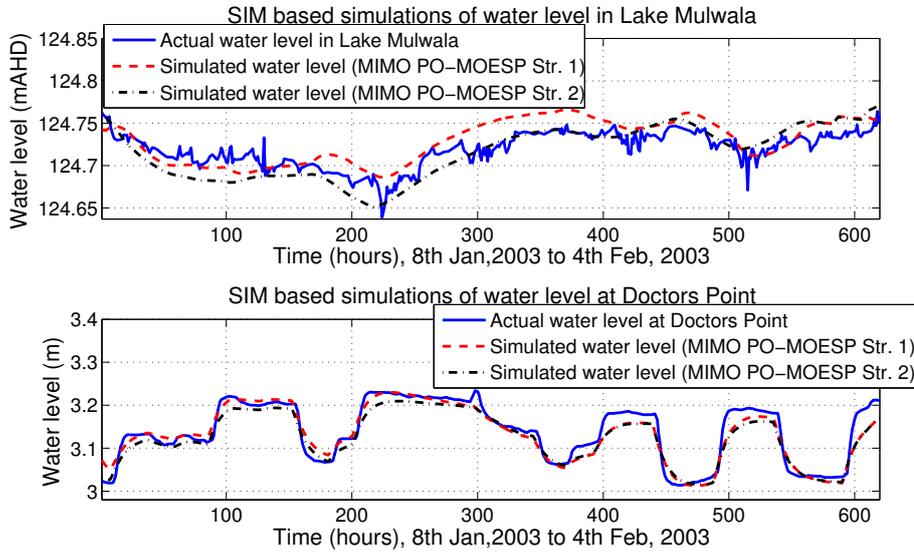


Figure 2.26: Simulation results on validation data.

$$B_{T'} = \begin{bmatrix} 0.003 & 0.007 & 0.018 & -0.014 & -0.021 & -0.016 \\ 0.086 & 0.094 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The signs of the entries in the first row of the $B_{T'}$ matrix reflect the physical reality as the entries associated with the inflows and outflows are positive and negative respectively. The signs of the entries (in A_T , B_T and C_T matrices) obtained in the first strategy also reflect the physical reality (easy to see when converted to transfer function representation), except the sign associated with the inflows from Bandiana appeared to be negative.

Figs. 2.26 & 2.27 show results of the MIMO PO-MOESP models on validation data. The simulation results are generally good, however, the model obtained from the second strategy did not perform as well against the water level in Lake Mulwala in Fig. 2.27 as the model obtained from the first strategy.

2.4 Comparison of results

In this section we continue the discussion from Section 2.2.6, and we compare the models and simulation results obtained in Section 2.3.

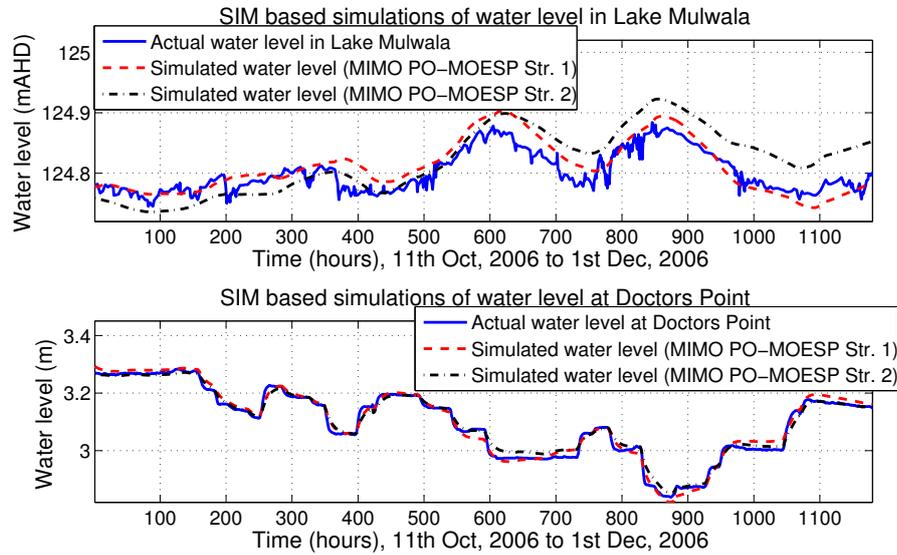


Figure 2.27: Simulation results on validation data.

Year	PEM	ML	CT-SS-PMF	CT-OE	SIM (Str. 1)	SIM (Str. 2)
LM (2003)	9.74×10^{-5}	1.82×10^{-4}	7.95×10^{-4}	9.98×10^{-5}	2.97×10^{-4}	2.63×10^{-4}
LM (2006)	3.81×10^{-4}	2.36×10^{-4}	1.37×10^{-4}	3.27×10^{-4}	4.62×10^{-4}	14.01×10^{-4}
DP (2003)	2.44×10^{-4}	2.18×10^{-4}	3.30×10^{-4}	2.43×10^{-4}	4.65×10^{-4}	5.52×10^{-4}
DP (2006)	2.52×10^{-4}	1.37×10^{-4}	5.32×10^{-4}	2.53×10^{-4}	2.59×10^{-4}	2.41×10^{-4}

Table 2.5: SSPE of water level simulations in Lake Mulwala and at Doctors Point.

2.4.1 Simulation performance and suitability for control

In Figs. 2.28 & 2.29 the simulation results of the identified models from Sections 2.3.1 – 2.3.4 are plotted together. The identified models pick up the trends in the validation data well. Almost all models behave similar with a few exceptions. The SIM based MIMO model (Strategy 2) did not simulate water level in Lake Mulwala against the validation dataset from 2006 as well as it did against the other dataset. Similarly, the CT SS-PMF model did not perform well against the validation dataset from 2003. This can be seen in Table 2.5 as well. The table gives the SSPEs (Eq. 2.32) corresponding to the simulation results in Figs. 2.28 & 2.29.

The identified models can be used in model based control techniques such as Model Predictive Control (MPC). To illustrate this, we used the MISO models of the water level

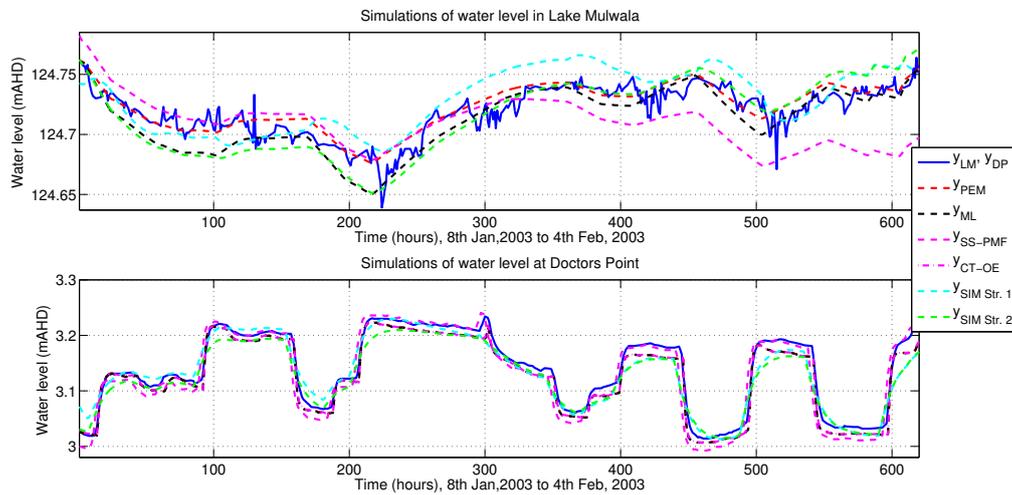


Figure 2.28: Simulation results on validation data.

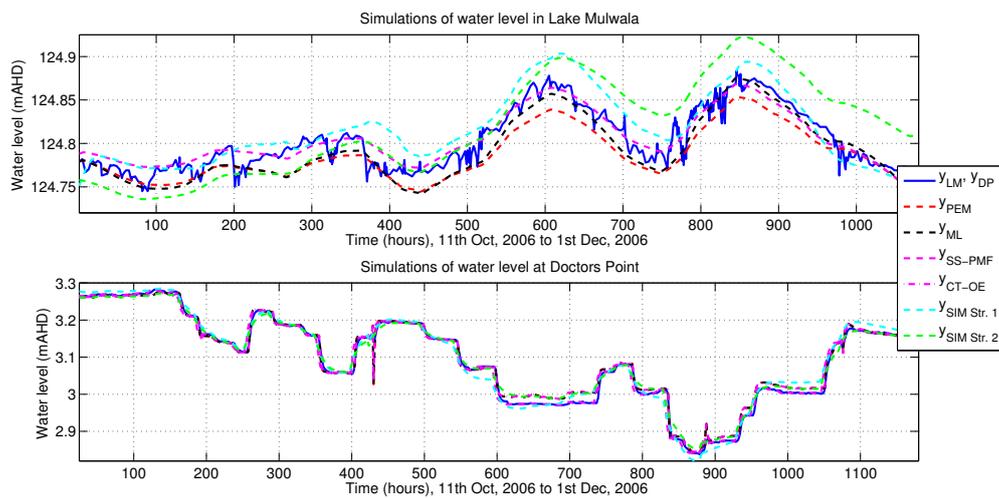


Figure 2.29: Simulation results on validation data.

Model	2002	2003	2004	2006
PEM (Avg. SSPE)	3.40×10^{-4}	1.35×10^{-4}	7.40×10^{-5}	2.95×10^{-5}
PEM (SSPE-LS)	8.30×10^{-4}	1.40×10^{-4}	1.15×10^{-4}	2.53×10^{-5}
CT-OE (Avg. SSPE)	3.42×10^{-4}	1.35×10^{-4}	7.36×10^{-5}	2.95×10^{-5}
CT-OE (SSPE-LS)	8.07×10^{-4}	1.40×10^{-4}	1.14×10^{-4}	2.52×10^{-5}
SRIVC (Avg. SSPE)	9.44×10^{-4}	2.09×10^{-4}	7.52×10^{-5}	6.27×10^{-5}
SRIVC (SSPE-LS)	18.01×10^{-4}	2.19×10^{-4}	7.82×10^{-5}	1.14×10^{-4}
SIM (Avg. SSPE)	4.22×10^{-4}	4.40×10^{-4}	2.35×10^{-4}	7.90×10^{-5}
SIM (SSPE-LS)	9.92×10^{-4}	11.01×10^{-4}	5.79×10^{-4}	1.72×10^{-4}

Table 2.6: Average SSPEs of 7 days simulation of water level in Lake Mulwala.

in Lake Mulwala obtained in Section 2.2, and simulated them for a horizon of 7 days starting from every hour (sample) of the four validation data used in Sections 2.2 & 2.3. Table 2.6 shows the average SSPEs corresponding to those simulations. In each case the SSPE is in the range of 10^{-5} to 10^{-4} . Table 2.6 also shows the sum of the squared prediction errors corresponding to the last simulation step (SSPE-LS) i.e. at the end of the 7th day. Again, the results are in the order of 10^{-4} on average i.e. the error in the water level after seven days open loop simulations is only a few centimetres. This is quite good taking into account the fact that the model contains an integrator such that error accumulates.

Fig. 2.30 shows the bode (magnitude) plots of the transfer functions from flows at Heywoods, Bandiana, Peechelba, (Downstream) Yarrawonga Weir, Yarrawonga Main Channel and Mulwala Canal to water level in Lake Mulwala (obtained in Section 2.2). The magnitude plots of the transfer functions obtained from PEM, CT-OE model, SRIVC and SIM are all close. The maximum difference between these models is about 6 dB except the magnitude plot of the SRIVC model displays a 12 dB lower gain for the transfer function from the flow at Bandiana as compared to the other models. Moreover, since the SRIVC model does not have poles at the origin (Table 2.2), the corresponding magnitude curves are approximately constant at very low frequencies (less than 10^{-6} rad/hour).

Fig. 2.31 shows the bode plots of the transfer functions from the flows at Heywoods and Bandiana to the flow at Doctors Point. The magnitude and the phase plots of the transfer functions of the PEM, ML and CT-OE models are all close. The plots of the SIM based model is different with the cut-off frequency nearly a decade lower. The slower

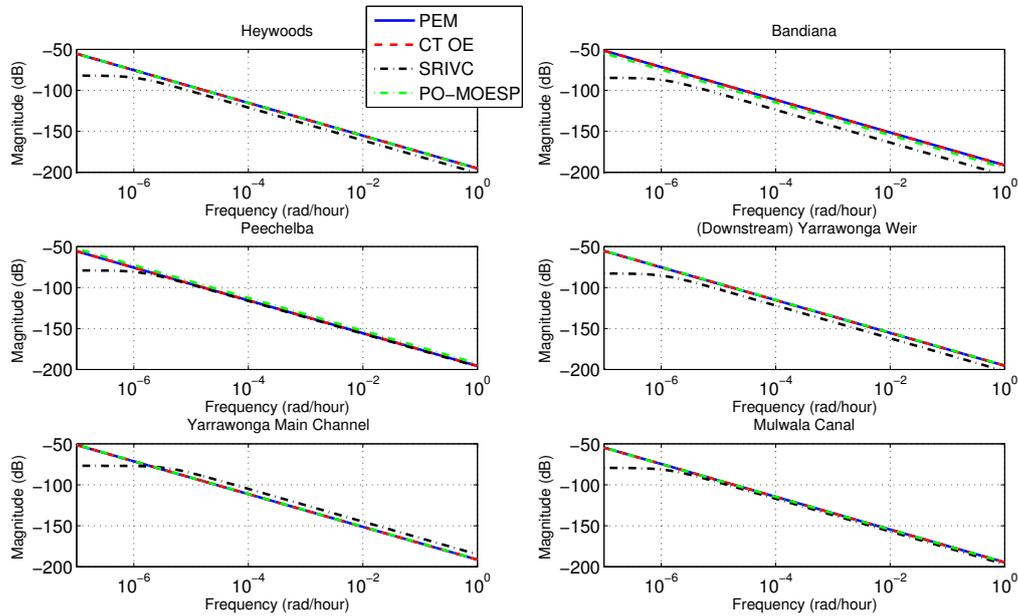


Figure 2.30: Transfer function plots from each input flows to water level in Lake Mulwala.

dynamics of the SIM model can also be seen in its simulation performance against the water level at Doctors Point in Fig. 2.26.

PEM, ML and CT-OE models gave consistently good performance in all aspects, and are in the authors' view good candidates as initial identification methods.

2.4.2 Prior knowledge

In optimisation based methods such as PEM and ML it is relatively easy to incorporate prior information about the direction of the flow by constraining parameters to be positive or negative, but it is difficult to do this in SIM and LS-PMF. Incorporation of known non-linearities are easy in methods which allows the user to freely select the model structure, but it becomes difficult in SIM. All methods considered can incorporate that lakes and reservoirs are often modelled as integrators either by explicitly including an integrator in the model structure or by considering $y(n) - y(n - 1)$ as the output of the system.

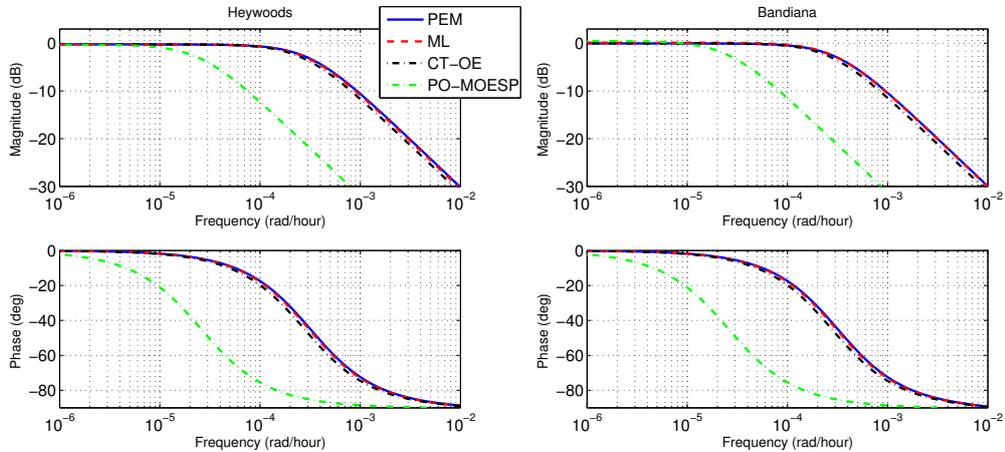


Figure 2.31: Transfer function plots from each input flows to flow at Doctors Point.

2.4.3 Model type

State space models are often preferred due to the availability of useful techniques for control, estimation and filtering for such models. SIM has an advantage in this respect as it produces MIMO state space models. The drawback is that there is usually no physical interpretation of the states as they are only given up to a similarity transform. In the case considered here it is easy to convert the obtained input-output models to state space form, and also, MIMO models can be constructed by combining the MISO models.

2.4.4 Computational and software issues

With optimisation based methods one can encounter difficulties finding a solution, while SIM is better in this regard since it is fast, non-iterative and has no local minima problems. However, no convergence issues or numerical issues were encountered in the optimisation based methods discussed in this paper.

This page intentionally left blank.

Appendix

2.A Additional figures

This appendix contains some additional figures which are referred in the main text, and they are for detailed analyses.

2.A.1 Flows of the validation data in Section 2.2

Figs. 2.32, 2.33, 2.34 & 2.35 show the flows at Heywoods, Bandiana, Peechelba, Downstream Yarrowonga Weir, Yarrowonga Main Channel and Mulwala Canal (Fig. 2.1), corresponding to the four validation data used in Section 2.2. The flows at Heywoods and Downstream Yarrowonga Weir are shown separately, because the flows at these locations are relatively higher than at other locations. The flow range of these validation data is almost similar to the flow range of the identification dataset in Fig. 2.2, except for the dataset in Fig. 2.34, where the flows at Heywoods and Mulwala Canal are relatively higher.

2.A.2 Simulation of the model in Eq. 2.6 with slight differences in τ_H

Slight differences in time delays do not affect the performance of the identified models much. We found that the estimated delays in this chapter could perform reasonably well with a change of time delays up to 3-4 hours.

To demonstrate this, we simulated water level in Lake Mulwala with different time delays from Heywoods, which is usually the largest of the in- and out-flows. We considered time delays of 67, 70 (the one used in this thesis) and 73 hours using the PEM-based

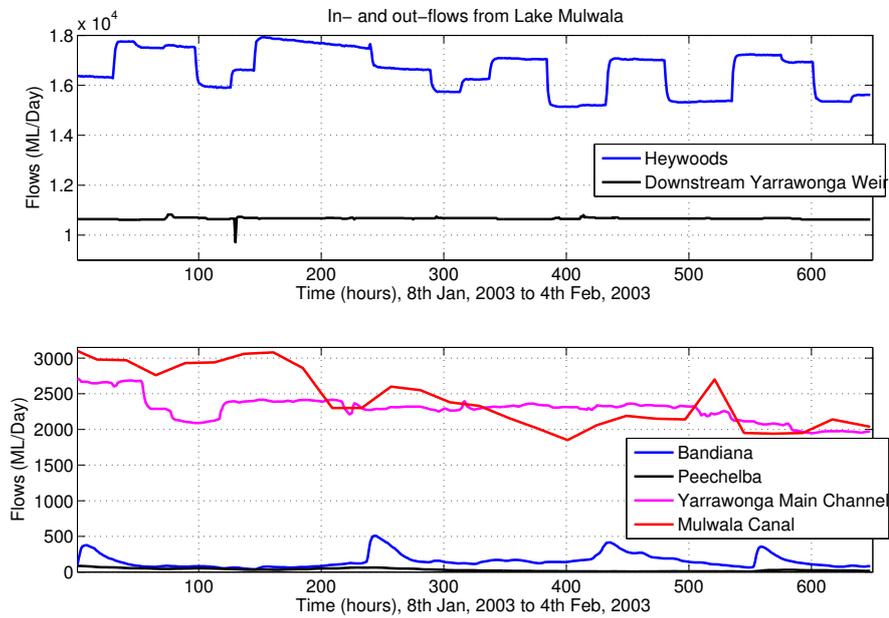


Figure 2.32: Validation dataset from year 2003.

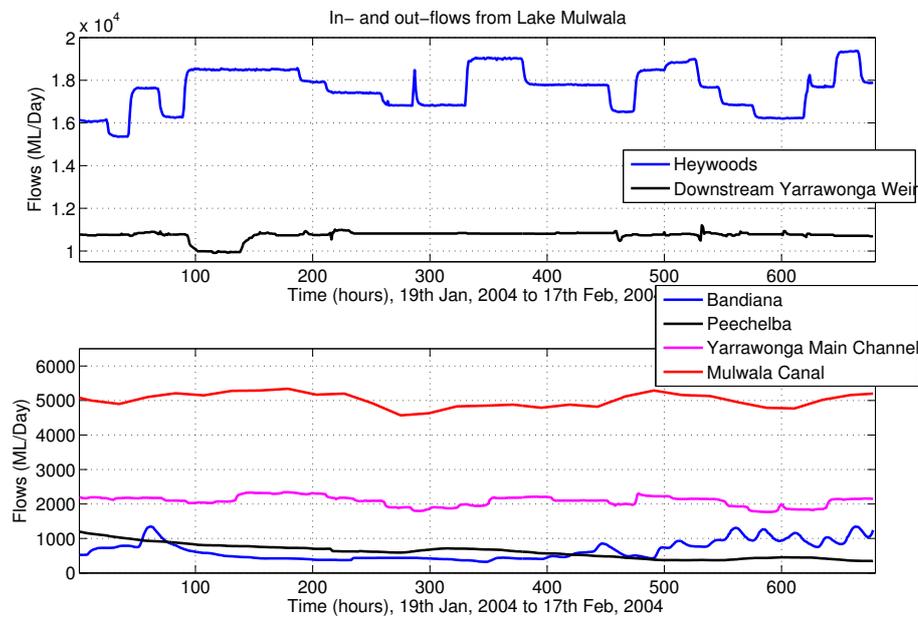


Figure 2.33: Validation dataset from year 2004.

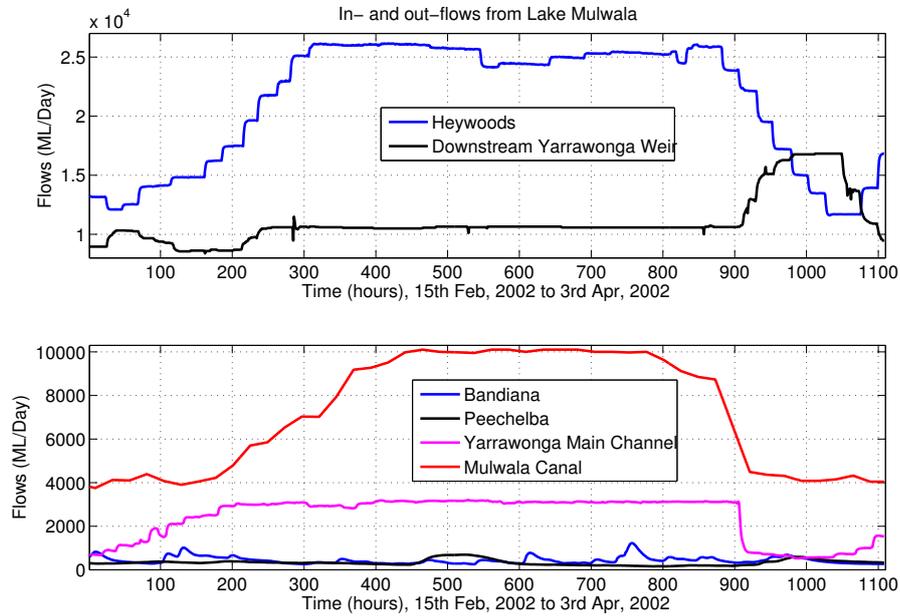


Figure 2.34: Validation dataset from year 2002.

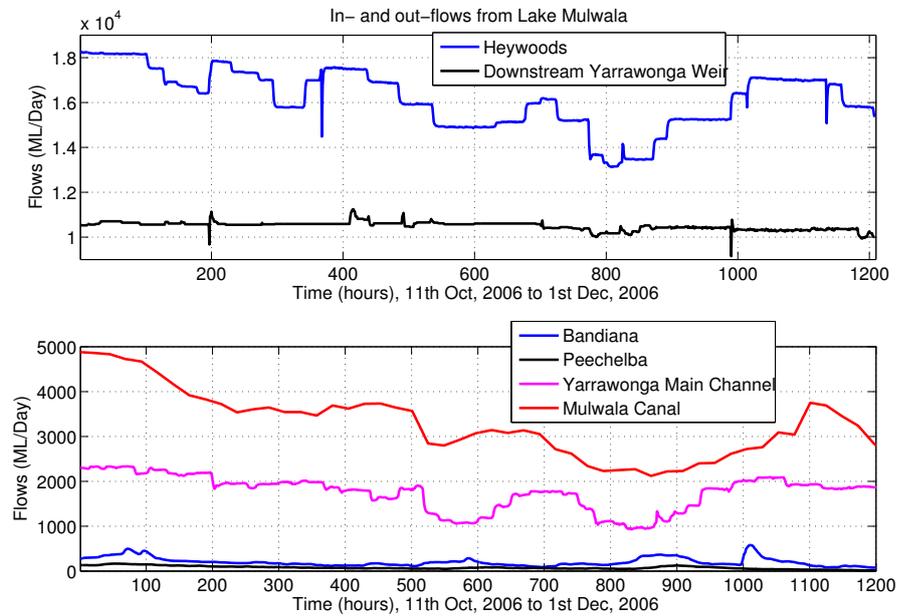


Figure 2.35: Validation dataset from year 2006.

estimated model in Section 2.2.1. Fig. 2.36 shows the simulation results. From the figure we can see that the simulation performance of the three models (with different time delays from Heywoods) is quite similar. Moreover, we can observe that there is hardly any difference in the performance for the first 300 hours (roughly 2 weeks).

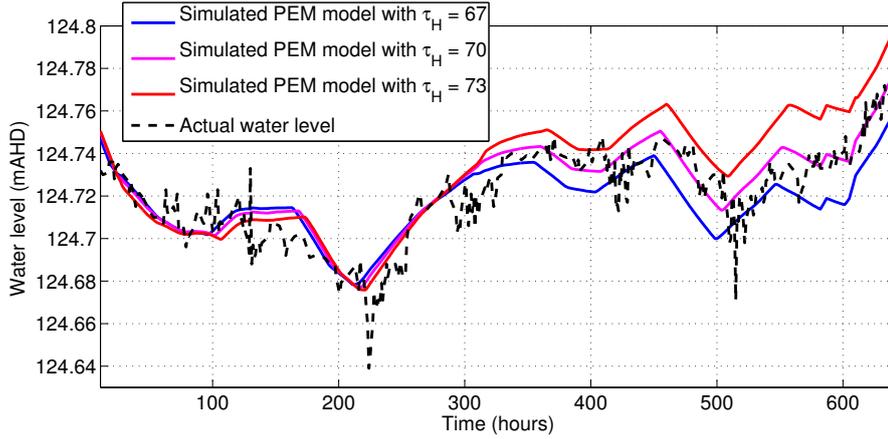


Figure 2.36: Simulation of PEM based model.

2.B DBM based higher order model of water level in Lake Mulwala

In this appendix we use Simplified Refined Instrumental Variable method for Continuous time systems (SRIVC), within the context of Data-Based Mechanistic (DBM) modelling, to identify a CT OE model of the water level in Lake Mulwala. However, different from Section 2.2.4, here we use the following CT OE model structure with extra parameter(s) in the numerator of the transfer function(s),

$$\begin{aligned}
 y_{LM}(t) = & \frac{b_H^1 s + b_H^0}{s + a_H} Q_H(t - \tau'_H) + \frac{b_B^1 s + b_B^0}{s + a_B} Q_B(t - \tau'_B) + \frac{b_P^1 s + b_P^0}{s + a_P} Q_P(t - \tau'_P) \\
 & + \frac{b_{DYW}^1 s + b_{DYW}^0}{s + a_{DYW}} Q_{DYW}(t) + \frac{b_{YMC}^1 s + b_{YMC}^0}{s + a_{YMC}} Q_{YMC}(t) + \frac{b_{MC}^1 s + b_{MC}^0}{s + a_{MC}} Q_{MC}(t).
 \end{aligned} \quad (2.45)$$

The parameters a_i and b_i^k are to be determined by the SRIVC method, where $i = \{H, B, P, DYW, YMC, MC\}$ and $k = \{0, 1\}$. We used the same time delays as used in Section 2.2.4, i.e. $\tau'_H = 70$, $\tau'_B = 71$ and $\tau'_P = 16$ hours. We have added a parameter in the

TF corresponding to flows	CT TFs
at Heywoods	$\frac{-9.48 \times 10^{-6}s + 3.39 \times 10^{-7}}{s + 0.0086} e^{-70s}$
at Bandiana	$\frac{-1.81 \times 10^{-5}s + 3.11 \times 10^{-8}}{s + 0.0034} e^{-71s}$
at Peechelba	$\frac{-1.20 \times 10^{-5}s + 8.37 \times 10^{-7}}{s + 0.0099} e^{-16s}$
over Yarrawoonga Weir	$\frac{-4.30 \times 10^{-6}s - 2.69 \times 10^{-7}}{s + 0.0097}$
to YW main channel	$\frac{5.71 \times 10^{-6}s - 2.49 \times 10^{-6}}{s + 0.029}$
to Mulwala Canal	$\frac{-3.20 \times 10^{-7}s - 4.65 \times 10^{-7}}{s + 0.0070}$

Table 2.7: CT transfer functions corresponding to each inflow and outflow from Lake Mulwala.

numerator to observe improvements in the validation results, compared to the results in Figs. 2.13 & 2.14, and to see whether it accommodates any missing dynamics.

We used the identification dataset in Fig. 2.2 and the CONTSID toolbox ([2]) to estimate the parameters in Eq. 2.45. The identified CT transfer functions (TFs) are shown in Table 2.7. The signs of the DC gains corresponding to each transfer function are in agreement with the in- and out-flows, i.e. they are positive for inflows and negative for outflows. The poles are in the left half plane and close to the origin, rather than at the origin. An interpretation of this in the DBM framework could be that there are small unaccounted-for losses in the system.

We validated the identified model against the same data as used in Section 2.2.4. Figs. 2.37 and 2.38 show the performance of the identified model. The model picks up the trends well against the data with small and relatively large variations, however, there is no visible improvement in the results compared to Figs. 2.13 & 2.14. The Sum of the Squared Prediction Errors (SSPEs) of the validation results in Figs. 2.37 & 2.38 were 2.16×10^{-4} , 3.77×10^{-4} and 2.80×10^{-3} , 3.68×10^{-4} respectively, which are (also) almost similar to the SSPEs corresponding to the results in Figs. 2.13 & 2.14, presented in Table 2.3. We also considered second order polynomials in the denominators of the model structure in Eq. 2.45, but the validation results got worse. Thus, these results validate that there is no need for a higher order model for this particular river stretch.

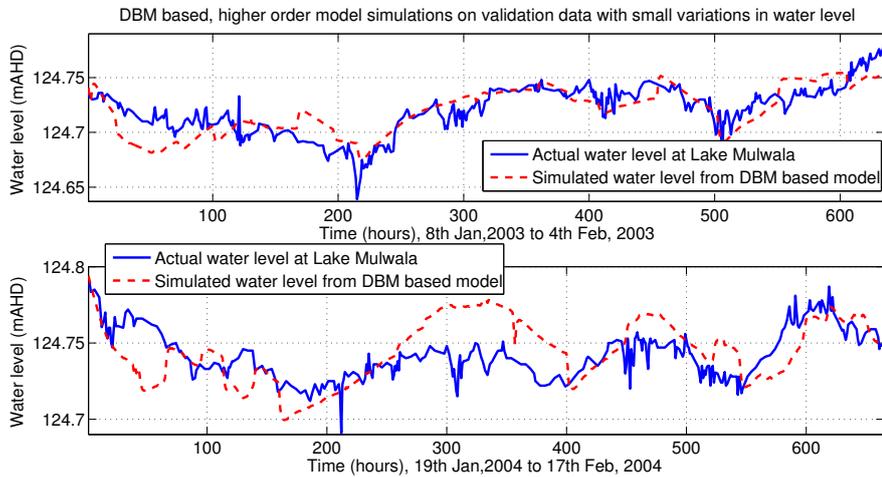


Figure 2.37: Simulation results on validation data with small variations in the water level in Lake Mulwala.

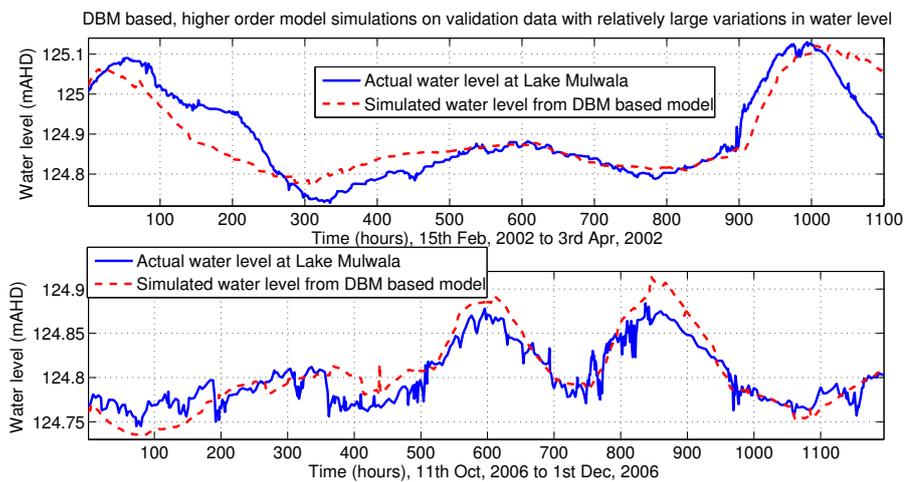


Figure 2.38: Simulation results on validation data with relatively large variations in the water level in Lake Mulwala.

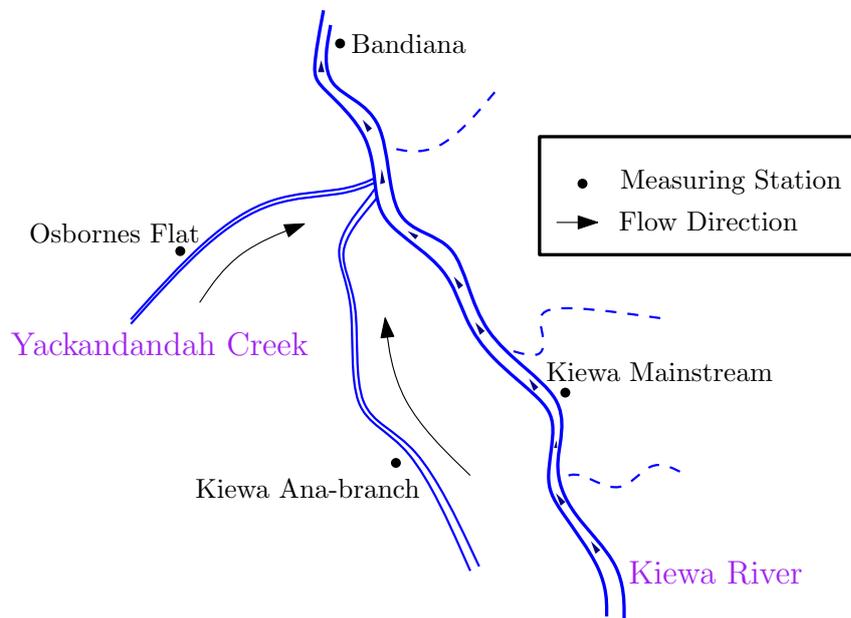


Figure 2.39: Kiewa River (plot not to scale).

2.C Kiewa River modelling

Kiewa River originates in the forest of Alpine National Park and Bogong High Planes in Victoria, Australia. It has several storages on its way e.g. Rocky Valley Storage, Lake Guy and Regulation Pondage. The storages have hydro-power plants and water release is regulated. Kiewa River has an ana-branch and it joins the main river from west. The ana-branch usually remains dry, however, when it rains, the branch contributes high peaks to the main river. Time delays along the ana-branch are larger because the channel gets wet first, and then water flows in it. There are several creeks that join Kiewa River, and Yackandandah Creek is one of the major creeks. It also contributes some high peaks to the main river. Time delays along the creek are smaller because the channel is straight and it usually remains wet. Some minor creeks enter Kiewa River from the mountains in the east, and the river experiences high flows in August and October.

We are interested in the upstream end of Kiewa River, especially the river stretch shown in Fig. 2.39, and we want to model flows at Bandiana. Based on the river sketch, flow at Bandiana is contributed by the inflows at Kiewa Mainstream (KMS), Kiewa Ana-branch (KAB) and Osbornes Flat (O). First, we consider the following model structure for

the flow at Bandiana,

$$Q_B(n) = c_{KMS}Q_{KMS}(n - \tau_{KMS}) + c_{KAB}Q_{KAB}(n - \tau_{KAB}) + c_OQ_O(n - \tau_O), \quad (2.46)$$

where the parameters $\theta_B = \{c_{KMS}, c_{KAB}, c_O\}$ and the time delays $\tau_B = \{\tau_{KMS}, \tau_{KAB}, \tau_O\}$ are unknown. An OE-type predictor can be derived from Eq. 2.46 as,

$$\hat{Q}_B(n|\theta_B, \tau_B) = c_{KMS}Q_{KMS}(n - \tau_{KMS}) + c_{KAB}Q_{KAB}(n - \tau_{KAB}) + c_OQ_O(n - \tau_O). \quad (2.47)$$

We used correlation co-efficient method to find the time delays. We considered low and high flow data, and on average we obtained the time delays as $\tau_{KMS} = 13$, $\tau_{KAB} = 19$ and $\tau_O = 5$ hours. An estimate of the parameter vector $\hat{\theta}_B$ can be obtained by solving the following optimisation problem,

$$\hat{\theta}_B = \arg \min_{\theta_B} \frac{1}{N - \tau_{KAB}} \sum_{n=\tau_{KAB}+1}^N (Q_B(n) - \hat{Q}_B(n|\theta_B, \tau_B))^2, \quad (2.48)$$

where N is the number of identification data-points. We used an identification dataset from year 2001, shown in Fig. 2.40. We can see that the major contribution to the flows at Bandiana (shown in the lower graph) is from the main stream of Kiewa River (shown in the upper graph). The flows in the ana-branch (Q_{KAB}) are approximately negligible in this dataset (the green curve in the upper graph, almost on the x-axis), but we cannot ignore this input, because as mentioned earlier, Q_{KAB} has contributed flow peaks in the past. We solved Problem (2.48) in MATLAB, and the parameters identified were as follows: $\hat{\theta}_B = \{0.99, 1.50, 1.17\}$, where the parameters were restricted between 0 and 1.5. We did this, because the parameter corresponding to the flows in the ana-branch (c_{KAB}) was picking a large value when it was not constrained, because the flows are negligible.

Figs. 2.41 & 2.42 show the simulations of the identified model against validation data from years 2003 and 2008, and the simulation results are good. There is a visible periodic behaviour in the flows at Bandiana in Fig. 2.42 between 2000th and 2500th sampling instants of the simulations. This is most likely due to a 24 hours release cycle from an upstream hydro electric power plant.

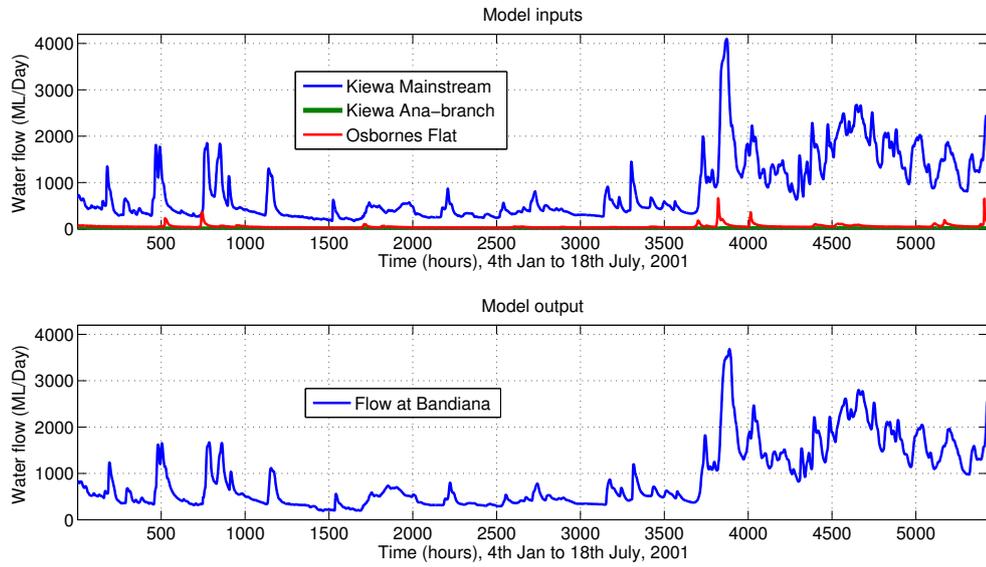


Figure 2.40: Identification dataset.

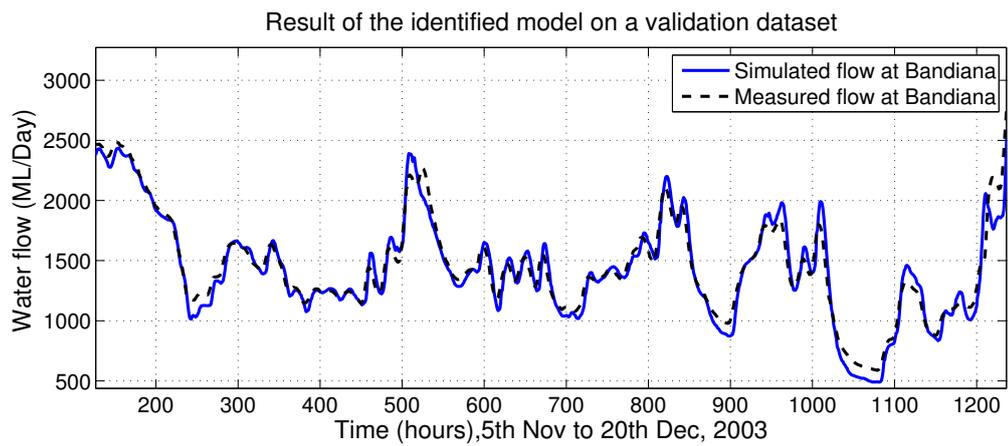


Figure 2.41: Simulation results on a validation dataset.

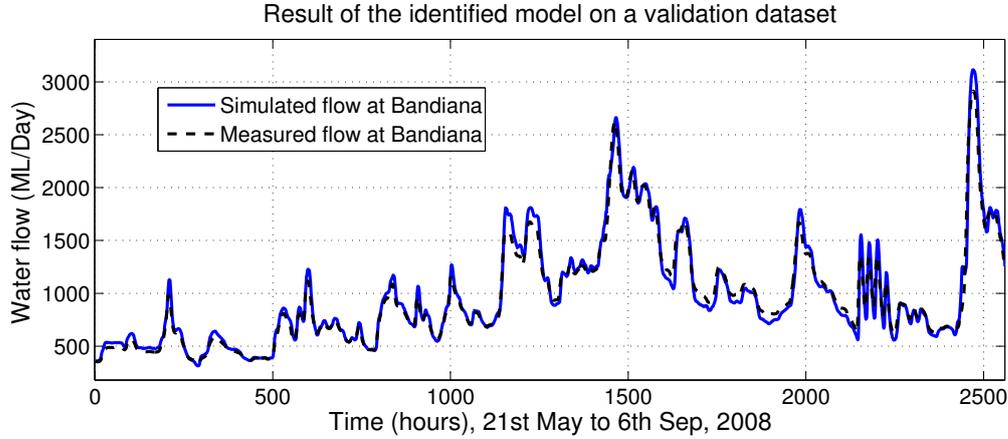


Figure 2.42: Simulation results on a validation dataset.

We also considered the following first order model structure for the flows at Bandiana,

$$Q_B(n+1) = c'_B Q_B(n) + c'_{KMS} Q_{KMS}(n - \tau_{KMS}) + c'_{KAB} Q_{KAB}(n - \tau_{KAB}) + c'_O Q_O(n - \tau_O). \quad (2.49)$$

The model structure in Eq. 2.49 is inspired from the model structure of the flow at Doctors Point, used in Section 2.3.1. We define the corresponding parameter vector as $\theta'_B = \{c'_B, c'_{KMS}, c'_{KAB}, c'_O\}$. An OE-type predictor can be derived from Eq. 2.49 as

$$\begin{aligned} \hat{Q}_B(n+1|\theta'_B, \tau_B) &= c'_B \hat{Q}_B(n|\theta'_B, \tau_B) + c'_{KMS} Q_{KMS}(n \\ &\quad - \tau_{KMS}) + c'_{KAB} Q_{KAB}(n - \tau_{KAB}) + c'_O Q_O(n - \tau_O), \end{aligned} \quad (2.50)$$

where $\tau_B = \{\tau_{KMS}, \tau_{KAB}, \tau_O\} = \{13, 19, 5\}$, as used before. Again, the parameters were estimated by solving the optimisation Problem (2.48), using the predictor in Eq. 2.50. We used the same identification dataset from 2001 (Fig. 2.40). The parameters identified were as follows: $\theta'_B = \{0.115, 0.880, 1, 1\}$, where the parameters were constrained between 0 and 1, because if they were kept unconstrained, they were picking negative or physically inexplicable values. Figs. 2.43 & 2.44 show the simulations of the identified model against the validation data from years 2003 and 2008, as used above. The simulation results are good and quite similar to the results shown in Figs. 2.41 & 2.42, and thus, both models are acceptable for this particular river stretch in Fig. 2.39.

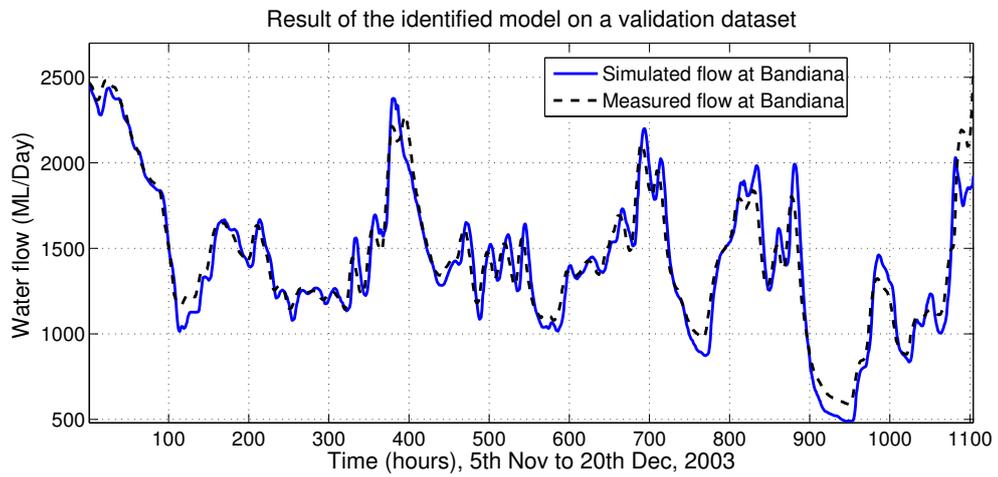


Figure 2.43: Simulation results on a validation dataset.

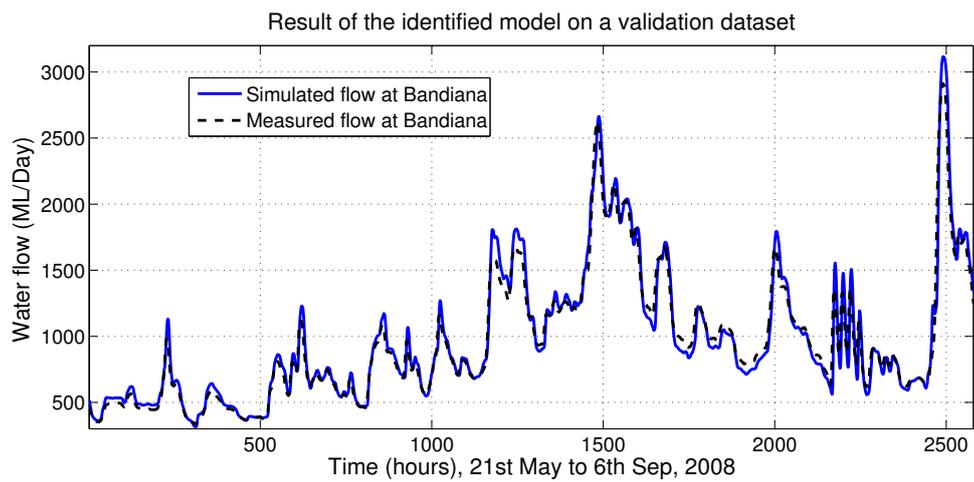


Figure 2.44: Simulation results on a validation dataset.

2.D Ovens River modelling

Ovens River flows into Murray River just upstream to Lake Mulwala (shown in Fig. 2.1). The river contributes as an unregulated inflow to the upper part of Murray River. The river experiences a lot of flow variations. In 2003, 2010 and 2012 Murray River had major flood events and one of the main reasons was the unregulated high inflow from Ovens River. Fig. 2.45 shows a rough sketch of the downstream end of the Ovens River. After Bright three rivers enter Ovens River: Buckland River, Buffalo River and King River. The flows in Buffalo River and King River are regulated at Lake Buffalo and Lake William Hovell respectively, and the inflows from these rivers are measured at Harris Lane and Cheshunt respectively. Fifteen Mile Creek enters Ovens River at Wangaratta, and the inflow is measured at Greta South. Several minor creeks also enter the river as shown in Fig. 2.45 with dashed lines. The combined flow in Ovens River is measured at Wangaratta and Peechelba, and from Peechelba there is a 28 km of river distance to Murray River. Also, the river has multiple ana-branches between Wangaratta and Peechelba (not shown in Fig. 2.45).

We want to model flow at Peechelba as it is a good indicator of the inflows from Ovens River to Murray River. From Fig. 2.45, we can see that the flow is contributed by the inflows at Bright (B), Harris Lane (H), Buffalo River (BR), Cheshunt (C) and Greta South (G). First, we propose the following model structure based on this information,

$$Q_P(n) = c_B Q_B(n - \tau_B) + c_H Q_H(n - \tau_H) + c_{BR} Q_{BR}(n - \tau_{BR}) + c_C Q_C(n - \tau_C) + c_G Q_G(n - \tau_G), \quad (2.51)$$

where the parameters $\theta_P = \{c_B, c_H, c_{BR}, c_C, c_G\}$ and time delays $\tau_P = \{\tau_B, \tau_H, \tau_{BR}, \tau_C, \tau_G\}$ are unknown. An OE-type predictor can be derived from Eq. 2.51 as,

$$\hat{Q}_P(n|\theta_P, \tau_P) = c_B Q_B(n - \tau_B) + c_H Q_H(n - \tau_H) + c_{BR} Q_{BR}(n - \tau_{BR}) + c_C Q_C(n - \tau_C) + c_G Q_G(n - \tau_G). \quad (2.52)$$

We used correlation coefficient method to calculate the time delays τ_P . We considered low and high flow data, and on average we got $\tau_B = 66$, $\tau_H = 67$, $\tau_{BR} = 63$, $\tau_C = 70$ and $\tau_G = 60$ hours. The estimates of the parameter vector $\hat{\theta}_P$ can be obtained by solving the

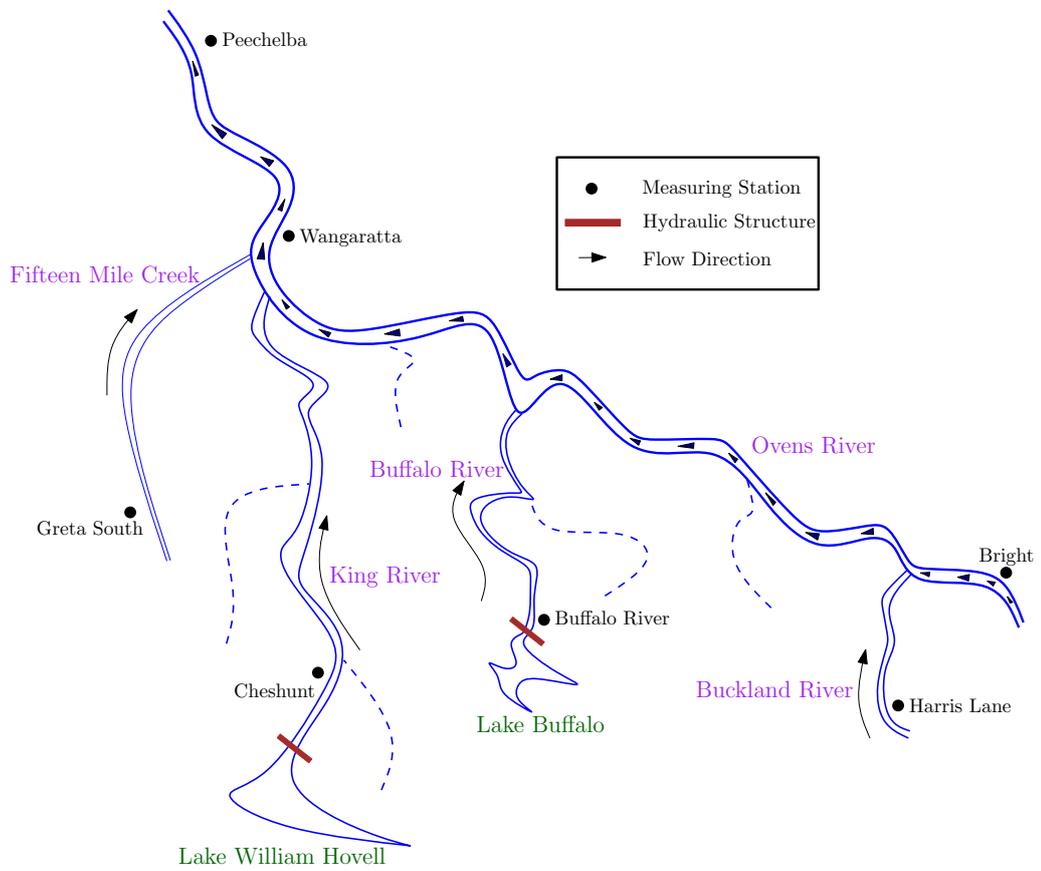


Figure 2.45: Ovens River (plot not to scale).

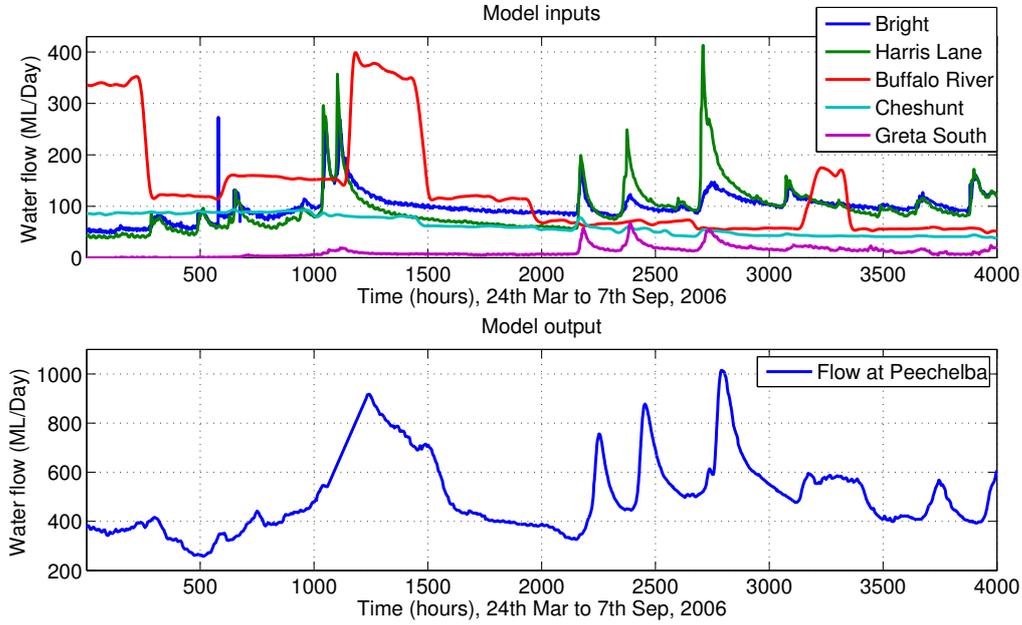


Figure 2.46: Identification dataset.

following optimization problem,

$$\hat{\theta}_P = \arg \min_{\theta_P} \frac{1}{N - \tau_C} \sum_{n=\tau_C+1}^N (Q_P(n) - \hat{Q}_P(n|\theta_P, \tau_P))^2, \quad (2.53)$$

where N is the number of identification data-points. We used an identification dataset from year 2006, shown in Fig. 2.46. Flows at Buffalo River and Cheshunt carry high frequency components, however, we are interested in the low frequency dynamics, so we lowpass filtered the flow measurements using a second order zero-phase Butterworth filter with cut off frequency $0.035 \text{ (hour)}^{-1}$. Fig. 2.46 shows filtered versions of the flows. We solved Problem (2.53), and the parameters identified were as follows $\hat{\theta}_B = \{1.5, 1.5, 0.67, 1.5, 1.5\}$, which were constrained between 0 and 1.5, because otherwise, they were picking some large values. The parameters were expected to be close to 1. The most likely explanation of this difference can be due to unaccounted-for non-linear effects of the multiple ana-branches and water pumping along the river. Figs. 2.47 & 2.48 show the simulation of the identified model against validation data from years 2003 and 2004. The identified model performed well against the data with different range of flows.

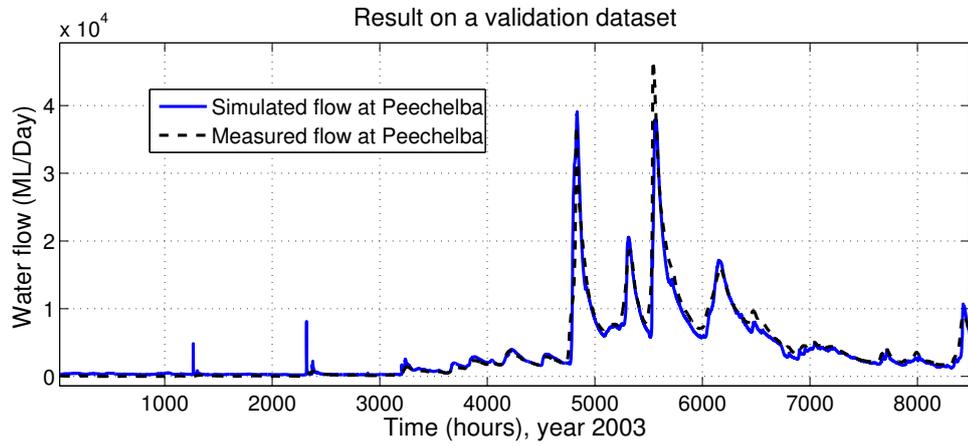


Figure 2.47: Simulation results on a validation dataset.

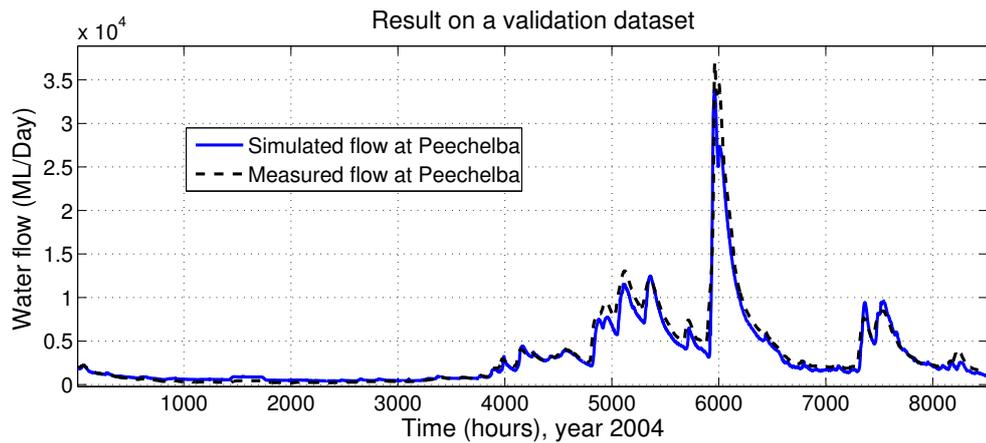


Figure 2.48: Simulation results on a validation dataset.

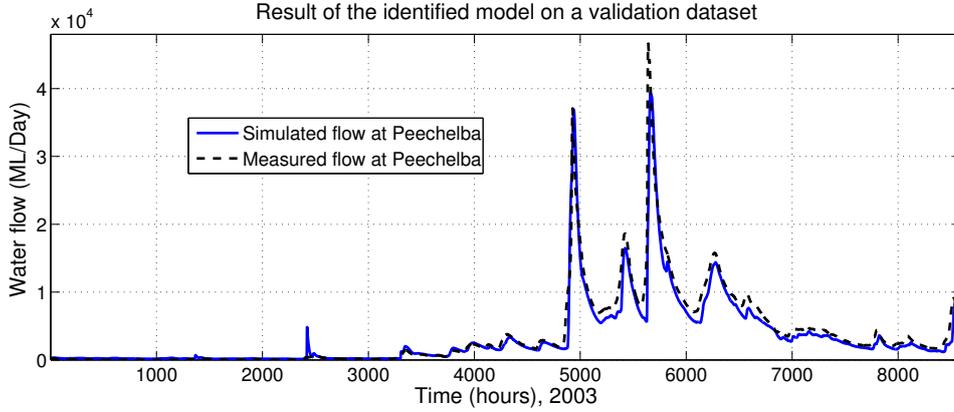


Figure 2.49: Simulation results on a validation dataset.

We also considered the following first order model structure for the flows at Peechelba,

$$\begin{aligned} Q_P(n+1) = & c'_P Q_P(n) + c'_B Q_B(n - \tau_B) + c'_H Q_H(n - \tau_H) \\ & + c'_{BR} Q_{BR}(n - \tau_{BR}) + c'_C Q_C(n - \tau_C) + c'_G Q_G(n - \tau_G). \end{aligned} \quad (2.54)$$

The model structure in Eq. 2.54 is inspired from the model structure of the flow at Doctors Point, used in Section 2.3.1. We define the corresponding parameter vector as $\theta'_P = \{c'_P, c'_B, c'_H, c'_{BR}, c'_C, c'_G\}$. An OE-type predictor can be derived from Eq. 2.54 as

$$\begin{aligned} \hat{Q}_P(n+1|\theta'_P, \tau_P) = & c'_P \hat{Q}_P(n|\theta'_P, \tau_P) + c'_B Q_B(n - \tau_B) + c'_H Q_H(n \\ & - \tau_H) + c'_{BR} Q_{BR}(n - \tau_{BR}) + c'_C Q_C(n - \tau_C) + c'_G Q_G(n - \tau_G), \end{aligned} \quad (2.55)$$

where $\tau_P = \{\tau_B, \tau_H, \tau_{BR}, \tau_C, \tau_G\} = \{66, 67, 63, 70, 60\}$, as used before. Again, the parameters were estimated by solving the optimisation Problem (2.53), using the predictor in Eq. 2.55. We used the same identification dataset from 2006 (Fig. 2.46). The parameters identified were as follows: $\theta'_B = \{0.926, 0.154, 0.076, 0.062, 0.010, 0.571\}$, where the parameters were constrained between 0 and 1, because if they were kept unconstrained, a few of them were picking negative values, which is physically inexplicable because all inputs are inflows. Figs. 2.49 & 2.50 show the simulations of the identified model against the validation data from years 2003 and 2004, as used above. The simulation results are good and quite similar to the results shown in Figs. 2.47 & 2.48, and thus, both models are acceptable for this particular river stretch in Fig. 2.45.

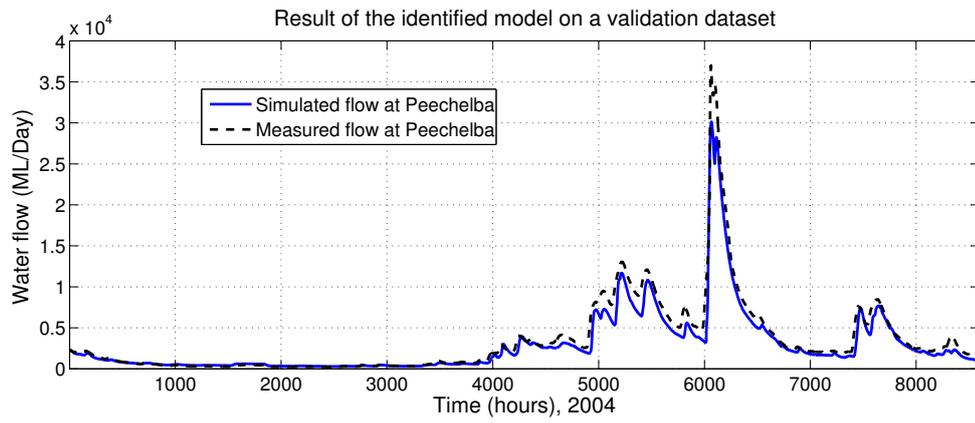


Figure 2.50: Simulation results on a validation dataset.

This page intentionally left blank.

Chapter 3

Scenario-based Stochastic Model Predictive Control (S-MPC) for Rivers

In Chapter 2 we considered different data-based methods to model rivers, and we identified models mainly for control purposes. We used the upper part of Murray River in Australia as a case study. In this chapter and in Chapter 5, we use river models to design controllers for the rivers, and we consider both normal river operations and flood operations. This chapter focuses on the normal river operations, and during these operations, river operators aim to achieve certain objectives, e.g. water level in a reservoir and flow release from the reservoir should be kept within safe limits. Similarly, the change in flows and water levels should be less than given thresholds. In this chapter we give a mathematical framework for control design for rivers, and we apply the developed control strategy to the upper part of Murray River in simulations.

Rivers have large distances between the locations where flows can be regulated and where controlled variables are measured. The large distances cause large time delays, and because of that, forecasts of the unregulated flows are required. Such forecasts are uncertain, and it makes the river control problem a challenging task [12]. Moreover, rivers have several physical and environmental constraints which are important and must be considered while designing controllers. Model Predictive Control (MPC) [22] has gained popularity in the past 15-20 years, and it can be considered as a natural choice in such situations, because of its ability to take constraints explicitly into account in the problem formulation. However, to incorporate the uncertainties, a Stochastic Model Predictive Control (S-MPC) based strategy can be considered, and it is explored in this thesis.

S-MPC requires knowledge of statistical properties of the uncertain elements in the

problem. A chance-constrained optimisation based control strategy can be used in such cases, where one can consider an average cost function with a set of probabilistic constraints, see e.g. [23–26]. In this strategy, violation of the constraints is accepted but for a few disturbance realisations only. However, the chance-constrained problems are generally non-convex and hence difficult to solve. Randomised strategies in [27–30, 82–85, 102] provide computationally tractable approximate solutions to such problems, especially the scenario approach introduced in [27–30, 102] is promising because of its simplicity. It is computationally tractable and it does not require any specific assumption on the nature of the disturbance, e.g. the distribution of the disturbance to be Gaussian or log concave etc.

For the above reasons, a scenario-based MPC approach suits the river control problem. However, feasibility of the optimisation problems (used in the MPC set-up) cannot be guaranteed, especially with tight constraint limits. To ensure feasibility, in this chapter, we present two schemes for river control problem where we relax the constraints. The first scheme is borrowed from [103], and formulated for rivers. The scheme provides a user chosen trade-off between performance and feasibility. The second scheme is developed in such a way that it does not require any user chosen parameter. Both schemes are applied to data from the upper part of Murray River in Australia.

The chapter is organised as follows. In Sections 3.1 & 3.2, we describe Model Predictive Control (MPC) and its stochastic versions for a general linear system. Sections 3.3 & 3.4 are dedicated to the formulation of the river control problem, where we develop a Stochastic MPC based control strategy for rivers. To solve the control problem, we propose scenario-based optimisation schemes in Section 3.5. The schemes ensure feasibility of the optimisation problem, and they are applied to the Murray River data in Section 3.6.

3.1 Model Predictive Control (MPC)

In this section we briefly describe Model Predictive Control (MPC), which is an on-line control technique. It solves an optimisation problem at each time step $n = 1, 2, \dots$, using a

predicted behaviour of the system over a finite horizon of M steps, and generates control sequences for the finite horizon, i.e. $\{u_n, u_{n+1}, \dots, u_{n+M-1}\}$, where $u \in \mathbb{R}^m$. Only the first control action, u_n , is implemented. At the next time step, $n + 1$, new measurements are obtained and the optimisation problem is solved again, and the process continues.

Next, we formulate an optimisation problem that can be solved in an MPC setting. Consider a linear state space system,

$$x(n+1) = Ax(n) + Bu(n), \quad (3.1)$$

$$y(n) = Cx(n), \quad (3.2)$$

where $x \in \mathbb{R}^{n_s}$ and $y \in \mathbb{R}^p$ are the states and outputs of the system respectively. For an objective criterion J , the optimisation problem at time n can be formulated as,

$$\begin{aligned} & \min_{\{u_n, u_{n+1}, \dots, u_{n+M-1}\}} J(u), \quad (3.3) \\ \text{s.t. } & x(i+1) = Ax(i) + Bu(i), \quad y_{\min} \leq Cx(i) \leq y_{\max}, \\ & u_{\min} \leq u(i) \leq u_{\max}, \quad \text{for } i = n, n+1, \dots, n+M-1, \end{aligned}$$

where i is a time index, u_{\min} , y_{\min} and u_{\max} , y_{\max} are the minimum and maximum allowed values of the inputs and outputs respectively. The constraints on $u(i)$ and $y(i)$ can also be described as $u(i) \in \mathcal{U}$ and $y(i) = Cx(i) \in \mathcal{Y}$, where \mathcal{U} and \mathcal{Y} are the sets to which u and y can belong, but we use the former formulation in this work. Quadratic cost functions are commonly used, and here we focus on the criterion functions of the type,

$$J = \sum_{i=n}^{n+M-1} [(x(i+1) - x_r)^\top Q' (x(i+1) - x_r) + u(i)^\top R' u(i) + \Delta u(i)^\top S' \Delta u(i)], \quad (3.4)$$

where x_r is a (constant) reference state vector, $\Delta u(i) = u(i) - u(i-1)$ is the change in control actions, Q' and R' are positive definite weighting matrices and S' is a positive semi-definite weighting matrix. With J as in Eq. 3.4, we minimise deviation of the states from their reference values, the control actions and the change in control actions in Problem 3.3, and the problem is a convex optimisation problem [104], provided

- J is convex function¹ with respect to the optimisation variables u .
- All equality and inequality constraints in the problem are convex in u .
- \mathcal{U} and \mathcal{Y} are convex sets², if $u(i) \in \mathcal{U}$ and $y(i) \in \mathcal{Y}$ are used as the constraints on inputs and outputs.

3.2 Stochastic MPC

In this section we describe stochastic versions of the MPC problem introduced in the previous section. We also describe how an approximate solution of a chance-constrained optimisation problem can be obtained in an MPC set-up.

As described earlier, MPC uses a model of the system to predict its future behaviour. In many applications, it is difficult to obtain an exact deterministic model, especially when there are external disturbances acting on the system which are difficult to describe. E.g. in rivers, unregulated flows are usually uncertain and it is difficult to encompass phenomenon like evaporation, rain and ground water-surface water interactions in a simple river model, aimed for control design purposes. In most of the cases, we model the system with an additional disturbance term $w \in \mathbb{R}^{n_s}$, such that Eqs. 3.1 & 3.2 become

$$x(n+1) = Ax(n) + Bu(n) + w(n), \quad (3.5)$$

$$y(n) = Cx(n). \quad (3.6)$$

In this thesis we assume the disturbances $w(n), w(n+1), \dots$ are independent and identically distributed. The literature provides two MPC approaches to control the above system: Robust MPC and Stochastic MPC. The two approaches depend on whether the disturbance term w can be bounded in a sensible way.

In Robust MPC [105–110], we assume that the disturbance term is bounded, and the MPC computes control actions that satisfy every possible disturbance realisation. Usu-

¹A function $f : \mathcal{X} \rightarrow \mathbb{R}$ is a convex function, if for all $x_1, x_2 \in \mathcal{X}$ and $\alpha \in [0, 1]$, then $f(\alpha x_1 + (1 - \alpha)x_2) \leq \alpha f(x_1) + (1 - \alpha)f(x_2)$.

²A set \mathcal{X} is said to be convex, if for all $x_1, x_2 \in \mathcal{X}$ and $\alpha \in [0, 1]$, then the point $\alpha x_1 + (1 - \alpha)x_2$ also belongs to \mathcal{X} .

ally a min-max approach [104] is considered, where we minimise the objective function against the worst possible disturbance realisation, and therefore the approach is considered conservative in this way, because it may significantly degrade the overall controller performance. Moreover, the distribution of the uncertain terms (disturbances) is not taken into account in this approach, and all disturbance realisations are treated as equally likely.

The literature also provides an alternative MPC approach to control stochastic systems, which is called Stochastic MPC (S-MPC) [23–26, 111, 112]. In this approach, $w(n)$ is a random variable, and the constraints are probabilistic constraints, which means that violations are accepted. However, they must happen for few disturbance realisations only. In this way S-MPC can avoid infeasibility of the constraints when the disturbance has an unbounded support. In S-MPC the objective criterion is the expected (average) value of the cost function in Eq. 3.4, i.e.

$$J_E = \mathbb{E} \left[\sum_{i=n}^{n+M-1} ((x(i+1) - x_r)^\top Q' (x(i+1) - x_r) + u(i)^\top R' u(i) + \Delta u(i)^\top S' \Delta u(i)) \right], \quad (3.7)$$

where \mathbb{E} is the conditional expectation over the disturbance terms given the initial state x_n . We assume that w belongs to a set \mathbf{W} , and a probability measure \mathbb{P}_W is defined on \mathbf{W} . Depending on the application, alternative formulations of the cost function can also be considered.

In this thesis, we consider S-MPC for the river control problem and we assume no bounded support on disturbances (which correspond to unregulated in- and outflows). We use the expected value of the cost function (as the criterion) and probabilistic constraints, and such problems are called Chance-Constrained optimisation Problems (CCPs) [23–26]. Probabilistic constraints are generally non-convex with respect to the optimisation variables, and thus a CCP is difficult to solve, since it does not lead to a convex optimisation problem.

Next, we formulate the control problem in a compact way. We introduce a feedback based control setting, and we formally state probabilistic constraints and the CCP problem. Also, we will describe an approximate method to find a solution of a CCP, which is

followed in this thesis.

3.2.1 State space representation for a finite time horizon

In this section we stack the states, outputs, control actions and disturbances of the system (Eqs. 3.5 & 3.6) over a finite horizon M , in vectors. The purpose is to form a compact representation of the system which will be used in the problem formulation ahead. Also, we use time indices in subscripts for compactness.

The relationship between the states at time n and $n+i$ can be obtained by using Eq. 3.5 in a recursive way,

$$x_{n+i} = A^i x_n + \begin{bmatrix} A^{i-1}B & \dots & B \end{bmatrix} \begin{bmatrix} u_n \\ \vdots \\ u_{n+i-1} \end{bmatrix} + \begin{bmatrix} A^{i-1} & \dots & I \end{bmatrix} \begin{bmatrix} w_n \\ \vdots \\ w_{n+i-1} \end{bmatrix}. \quad (3.8)$$

For a finite time horizon M , let the vectors of states, control actions, disturbances and outputs be $\mathbf{x}_{n+1} = [x_{n+1} \ x_{n+2} \ \dots \ x_{n+M}]^\top$, $\mathbf{u}_n = [u_n \ u_{n+1} \ \dots \ u_{n+M-1}]^\top$, $\mathbf{w}_n = [w_n \ w_{n+1} \ \dots \ w_{n+M-1}]^\top$ and $\mathbf{y}_n = [y_n \ y_{n+1} \ \dots \ y_{n+M-1}]^\top$ respectively, and \mathbf{w}_n is assumed to belong to a set $\mathbf{W} = \mathbf{W}^M$.

Using Eq. 3.8 with $i = 1, 2, \dots, M$, we obtain the following model for \mathbf{x}_{n+1} ,

$$\mathbf{x}_{n+1} = F\mathbf{x}_n + G\mathbf{u}_n + H\mathbf{w}_n, \quad (3.9)$$

with

$$F = \underbrace{\begin{bmatrix} A \\ A^2 \\ \vdots \\ A^M \end{bmatrix}}_{(Mn_s \times n_s)}, \quad G = \underbrace{\begin{bmatrix} B & \mathbf{0} & \dots & \mathbf{0} \\ AB & B & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ A^{M-1}B & A^{M-2}B & \dots & B \end{bmatrix}}_{(Mn_s \times Mm)}, \quad H = \underbrace{\begin{bmatrix} I & \mathbf{0} & \dots & \mathbf{0} \\ A & I & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ A^{M-1} & A^{M-2} & \dots & I \end{bmatrix}}_{(Mn_s \times Mn_s)},$$

where n_s is the number of states and m is the number of control inputs. Also, from Eq. 3.6, \mathbf{y}_n is given by,

$$\mathbf{y}_n = C\mathbf{x}_n, \quad (3.10)$$

where $\mathbf{C} = \underbrace{\begin{bmatrix} C & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & C & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & C \end{bmatrix}}_{(Mp \times Mn_s)}$, and C is the output matrix in Eq. 3.6.

3.2.2 Feedback based control policy

Problem (3.3) generates a sequence of control actions over the finite horizon by optimising the criterion with respect to control values. For stochastic systems, it is advisable to optimise the criterion over control policies rather than control values in order to incorporate more feedback related to the the disturbances acting on the system. In this section we define a feedback based control policy for the system in Eq. 3.9.

A state feedback approach is a natural choice of control policy. Assuming the states x_n are available at time n , then we can generate control inputs at time n as

$$u_n = Kx_n + g_n, \quad (3.11)$$

where K is typically chosen off-line and g_n can be selected on-line [113], [114], [115] & [116]. However, a single value of K , chosen off-line, might not work for applications where the constraints in the system affect the set of allowed K values. We can use a time-varying affine state feedback control policy as $u_n = K_n x_n + g_n$, and with the knowledge of prior states, we can modify and improve the control policy as [115]

$$u_n = \sum_{i=0}^n K_{n,i} x_i + g_n, \quad (3.12)$$

where $K_{n,i} \in \mathbb{R}^{m \times n_s}$ and $g_n \in \mathbb{R}^m$. Thus, with such a control policy, the vector of control actions \mathbf{u}_n can be described as

$$\mathbf{u}_n = \mathbf{K}_n \mathbf{x}_n + \mathbf{g}_n, \quad (3.13)$$

where

$$\mathbf{K}_n = \underbrace{\begin{bmatrix} K_{n,0} & \mathbf{0} & \dots & \mathbf{0} \\ K_{n+1,0} & K_{n+1,1} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ K_{n+M-1,0} & K_{n+M-1,1} & \dots & K_{n+M-1,M-1} \end{bmatrix}}_{(Mm \times Mn_s)}, \quad \mathbf{g}_n = \underbrace{\begin{bmatrix} g_n \\ g_{n+1} \\ \vdots \\ g_{n+M-1} \end{bmatrix}}_{(Mm \times 1)}.$$

However, if we substitute Eq. 3.13 in Eq. 3.9 we get

$$\mathbf{x}_n = (qI - G\mathbf{K}_n)^{-1}(F\mathbf{x}_n + G\mathbf{g}_n + H\mathbf{w}_n),$$

where q is the time shift operator, and the control actions from Eq. 3.13 becomes,

$$\mathbf{u}_n = \mathbf{K}_n(qI - G\mathbf{K}_n)^{-1}(F\mathbf{x}_n + G\mathbf{g}_n + H\mathbf{w}_n) + \mathbf{g}_n.$$

The above equations give a non-linear mapping from \mathbf{K}_n and \mathbf{g}_n to \mathbf{x}_n and \mathbf{u}_n , because \mathbf{K}_n appears in an inverse matrix expression. Hence, optimisation over \mathbf{K}_n and \mathbf{g}_n is likely to be difficult, since the parametrisation is non-convex in \mathbf{K}_n . Thus, it cannot be incorporated in a standard convex optimisation problem [115]. The same conclusions are drawn in [114] with a slightly different interpretation.

Alternatively, the control policies can be parametrised as an affine function of past disturbances [114, 115], which are linearly related to the current states (Eqs. 3.5 & 3.9). By (recursively) using Eq. 3.5 in the state feedback policy $u_n = K_n x_n$, we get the control policy in terms of initial states and past disturbances as

$$u_n = K_n \prod_{i=0}^{n-1} (A + BK_i) x_0 + \sum_{j=1}^{n-1} K_n \prod_{i=j}^{n-1} (A + BK_i) w_{j-1} + K_n w_{n-1}. \quad (3.14)$$

Such a control policy can be parametrised as

$$u_n = \gamma_n + \theta_{n,0} w_0 + \theta_{n,1} w_1 + \dots + \theta_{n,n-1} w_{n-1}, \quad (3.15)$$

where $\theta_{n,i} \in \mathbb{R}^{m \times n_s}$ and $\gamma_n \in \mathbb{R}^m$. For a finite horizon M , a vector of control policies is

given by

$$\mathbf{u}_n = \Theta_n \mathbf{w}_n + \Gamma_n, \quad (3.16)$$

where

$$\Gamma_n = \begin{bmatrix} \gamma_n \\ \gamma_{n+1} \\ \vdots \\ \underbrace{\gamma_{n+M-1}}_{(Mm \times 1)} \end{bmatrix}, \quad \Theta_n = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \theta_{n+1,0} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \underbrace{\theta_{n+M-1,0} \quad \theta_{n+M-1,1} \quad \dots \quad \mathbf{0}}_{(Mm \times Mn_s)} \end{bmatrix}. \quad (3.17)$$

If we substitute Eq. 3.16 in Eq. 3.9 we get

$$\mathbf{x}_n = Fx_n + G\Gamma_n + (G\Theta_n + H)\mathbf{w}_n.$$

In the above equation and in Eq. 3.16, the mapping from Θ_n and Γ_n to \mathbf{x}_n and \mathbf{u}_n is linear. Hence, this choice of parametrisation leads to a convex problem formulation [114, 115]. The control policy also ensures causality as the control action at time n is defined in terms of the disturbances at time $n - 1$ and earlier.

To use Eq. 3.16 as a control policy, we need, at time n , the value of w_{n-1} (see Eq. 3.15), which can be obtained from the state measurements at time n using Eq. 3.5

$$w_{n-1} = x_n - Ax_{n-1} - Bu_{n-1}. \quad (3.18)$$

The parametrisation in Eq. 3.17 has in-total $mM + mn_s M(M - 1)/2$ parameters. This parametrisation is applicable if the whole state vector x_n is measurable. However, when the whole state vector is not measurable, then the parametrisation requires slight modifications. Effectively, the parameter entries in $\theta_{i,j}$ s (in the Θ_n matrix)—corresponding to the disturbances which are not computable—are set to zero, and this does not affect the convexity results discussed above.

There are other parametrisation rules for Γ_n and Θ_n matrices (in Eq. 3.16) suggested in [27], e.g.

1. Θ_n only has k non-zero sub-diagonals, where $k < M - 1$,

$$\Gamma_n = \begin{bmatrix} \gamma_n \\ \gamma_{n+1} \\ \vdots \\ \gamma_{n+k} \\ \vdots \\ \gamma_{n+M-1} \end{bmatrix}, \Theta_n = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \dots & \dots & \mathbf{0} & \mathbf{0} \\ \theta_{n+1,0} & \mathbf{0} & \dots & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \dots & \dots & \vdots & \vdots \\ \theta_{n+k,0} & \ddots & \ddots & \dots & \vdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \theta_{n+M-1,M-1-k} & \dots & \theta_{n+M-1,M-1} & \mathbf{0} \end{bmatrix}.$$

The number of parameters d in this parametrisation are

$$d = mM + mn_s(k(k-1)/2 + k(M-1-k)).$$

In comparison with the parametrisation in Eq. 3.17, here the Θ_n matrix has $mn_s(M(M-1-2k) + k(k+3))/2$ less parameters, which is computationally better, when used in an optimisation problem. In this parametrisation, the parameters in the Θ_n matrix, corresponding to disturbances w_n in Eq. 3.16, are different for different time instants.

2. We can further reduce the number of parameters by using fixed parameters on the sub-diagonals of the Θ_n matrix in Eq. 3.17,

$$\Gamma_n = \begin{bmatrix} \gamma_n \\ \gamma_{n+1} \\ \vdots \\ \gamma_{n+M-2} \\ \gamma_{n+M-1} \end{bmatrix}, \Theta_n = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \theta_{n+1} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \dots & \vdots & \vdots \\ \theta_{n+M-2} & \ddots & \ddots & \vdots & \vdots \\ \theta_{n+M-1} & \theta_{n+M-2} & \dots & \theta_{n+1} & \mathbf{0} \end{bmatrix}. \quad (3.19)$$

In this case the number of parameters d are,

$$d = mM + mn_s(M-1). \quad (3.20)$$

In this parametrisation, the parameters in the Θ_n matrix, corresponding to the con-

control variables u_i and disturbances $w_{i-\delta}$ in Eq. 3.16, for $i = n+1, n+2, \dots, n+M-1$ and $\delta = 1, 2, \dots, i-n$, are kept the same. It means that the parameters corresponding to the disturbances and control variables which are δ time steps away from each other are kept the same. In this thesis, in order to reduce the computational burden, we will use this parametrisation in the river control problem.

3.2.3 Probabilistic constraints and Chance-Constrained optimisation Problem (CCP)

In this section we describe different kinds of probabilistic constraints and then state a CCP problem which can be solved in a Stochastic MPC setting.

A hard constraint on the state vector \mathbf{x}_n (in Eq. 3.9) or on the output vector \mathbf{y}_n (in Eq. 3.10) can lead to infeasibilities. The reason for the infeasibilities is that the disturbance vector \mathbf{w}_n is additive in Eq. 3.9, and it may have unbounded support. To avoid this, we consider soft (probabilistic) constraints, which means that some violations of the constraint are allowed. However, such probabilistic constraints must hold on the set of disturbance realisations \mathbb{W} , with probability at least $1 - \epsilon$, where $\epsilon \in (0, 1)$ is a user chosen parameter. The disturbance vector \mathbf{w}_n also appears multiplicatively in the feedback control policy (Eq. 3.16), but a hard constraint on the control input vector does not lead to infeasibility, because the Θ_n matrix in Eq. 3.16 can be set to zero. This leads to the following two categories of constraints setting [103],

- ‘Hard and soft’ i.e. hard constraints on inputs and soft constraints on state variables. Now, we consider the constraints on the input and output in Problem (3.3) over the finite horizon M . Let $\|\cdot\|_\infty$ be the infinity norm, which gives the maximum value of a variable or a vector. We use the infinity norm on the state and control input vectors to define the constraints in Problem (3.3) in a compact and ‘hard and soft’ format as below,

$$u_{\min} \leq \|\mathbf{u}_n(\mathbf{w}_n)\|_\infty \leq u_{\max},$$

$$\mathbb{P}\{\mathbf{w}_n \in \mathbb{W} : y_{\min} \leq \|\mathbf{C}\mathbf{x}_n(\mathbf{w}_n)\|_\infty \leq y_{\max}\} \geq 1 - \epsilon.$$

- ‘Soft and soft’ i.e. soft constraints on both inputs and state variables i.e.

$$\mathbb{P}\{\mathbf{w}_n \in \mathbb{W} : u_{\min} \leq \|\mathbf{u}_n(\mathbf{w}_n)\|_{\infty} \leq u_{\max} \cap y_{\min} \leq \|\mathbf{C}\mathbf{x}_n(\mathbf{w}_n)\|_{\infty} \leq y_{\max}\} \geq 1 - \epsilon, \quad (3.21)$$

where $\mathbb{P} = \mathbb{P}_{\mathbb{W}}^M$ denotes the probability measure on the set of disturbance \mathbb{W} . In many applications we also constrain the change of inputs, $\mathbf{u}_n(\mathbf{w}_n) - \mathbf{u}_{n-1}(\mathbf{w}_{n-1}) = \Delta\mathbf{u}_n(\mathbf{w}_n)$, as $\Delta u_{\min} \leq \|\Delta\mathbf{u}_n(\mathbf{w}_n)\|_{\infty} \leq \Delta u_{\max}$, where Δu_{\min} and Δu_{\max} are the lower and upper limits on the change of inputs. This constraint can also be considered as hard or soft, however, in this thesis we consider all constraints as soft constraints.

Using the above constraints and the objective function J_E from Eq. 3.7 we define a CCP which can be solved in an S-MPC setting,

$$\begin{aligned} \min_{\Theta, \Gamma} \quad & \mathbb{E}[(\mathbf{x}_{n+1} - \mathbf{x}_r)^\top \mathbf{Q}(\mathbf{x}_{n+1} - \mathbf{x}_r) + \mathbf{u}_n^\top \mathbf{R}\mathbf{u}_n + \Delta\mathbf{u}_n^\top \mathbf{S}\Delta\mathbf{u}_n], \\ \text{s.t.} \quad & \mathbb{P}\{\mathbf{w}_n \in \mathbb{W} : u_{\min} \leq \|\mathbf{u}_n(\mathbf{w}_n)\|_{\infty} \leq u_{\max} \cap \Delta u_{\min} \leq \|\Delta\mathbf{u}_n(\mathbf{w}_n)\|_{\infty} \leq \Delta u_{\max} \cap \\ & y_{\min} \leq \|\mathbf{C}\mathbf{x}_n(\mathbf{w}_n)\|_{\infty} \leq y_{\max}\} \geq 1 - \epsilon, \end{aligned} \quad (3.22)$$

subject to the system dynamics, \mathbf{x}_{n+1} in Eq. 3.9, and the control policy, \mathbf{u}_n in Eq. 3.16. \mathbf{Q} , \mathbf{R} and \mathbf{S} are block diagonal matrices with the \mathbf{Q}' , \mathbf{R}' and \mathbf{S}' matrices (Eq. 3.7) on the block diagonals respectively. \mathbf{x}_r is a vector with M reference state vectors x_r (Eq. 3.7) stacked together. In a Stochastic MPC setting, we solve Problem (3.22) in a receding horizon fashion. However, the problem is difficult to solve, because in general, a probabilistic constraint is non-convex with respect to the optimisation variables³. E.g. $\mathbb{P}\{\mathbf{w}_n \in \mathbb{W} : u_{\min} \leq \|\mathbf{u}_n(\mathbf{w}_n)\|_{\infty} \leq u_{\max}\} \geq 1 - \epsilon'$, in general, is non-convex with respect to the optimisation variables (Θ and Γ), even if $u_{\min} \leq \|\mathbf{u}_n(\mathbf{w}_n)\|_{\infty} \leq u_{\max}$ is convex in Θ and Γ .

On the other hand, the randomised strategies in [27–30, 82–85, 102] provide computationally tractable approximate solutions to CCPs. We use the scenario approach [27–30, 102] in this thesis, and it is described in the next section.

³There are some cases when a CCP is convex, e.g. when \mathbf{w}_n is normally distributed with linear inequality constraints in the problem, or when the distribution of \mathbf{w}_n is log-concave [104].

3.2.4 Scenario-based approach to CCPs

As mentioned in the previous sub-section, it is hard to solve CCPs. To find an approximate solution to a CCP, we employ a scenario based randomised approach [27–30, 102]. Here, we first illustrate the idea of the approach using Problem (3.22), before stating the scenario theorem.

To find a scenario based approximate solution of Problem (3.22), we generate N_r realisations of the disturbance \mathbf{w}_n according to a given probability distribution, which is assumed to be available (i.e. the probability measure \mathbb{P} in Problem (3.22) is assumed to be known), and then we replace Problem (3.22) with

$$\begin{aligned} \min_{\Theta, \Gamma} \quad & \mathbb{E}[(\mathbf{x}_{n+1} - \mathbf{x}_r)^\top \mathbf{Q}(\mathbf{x}_{n+1} - \mathbf{x}_r) + \mathbf{u}_n^\top \mathbf{R} \mathbf{u}_n + \Delta \mathbf{u}_n^\top \mathbf{S} \Delta \mathbf{u}_n], \\ \text{s.t.} \quad & u_{\min} \leq \|\mathbf{u}_n(\mathbf{w}_n^{(k)})\|_\infty \leq u_{\max}, \quad \Delta u_{\min} \leq \|\Delta \mathbf{u}_n(\mathbf{w}_n^{(k)})\|_\infty \leq \Delta u_{\max}, \\ & y_{\min} \leq \|\mathbf{C} \mathbf{x}_n(\mathbf{w}_n^{(k)})\|_\infty \leq y_{\max}, \quad \text{for } k = 1, 2, \dots, N_r, \end{aligned} \quad (3.23)$$

i.e. we replace every probabilistic constraint in Problem (3.22) with N_r constraints, each corresponding to an independent realisation of the disturbance vector \mathbf{w}_n . Problem (3.23) is a *scenario problem*, and it is a computationally tractable approximation of Problem (3.22). Moreover, according to the scenario theorem ([117]) stated below, the scenario solution provides with high confidence a feasible solution to the chance-constrained Problem (3.22), provided N_r is chosen large enough.

Theorem 3.1. *If the number of scenarios N_r used in a scenario problem satisfies*

$$\sum_{i=0}^{d-1} \binom{N_r}{i} \epsilon^i (1 - \epsilon)^{N_r - i} \leq \beta, \quad (3.24)$$

where $\epsilon \in (0, 1)$, $\beta \in (0, 1)$ and d is the number of optimisation variables, then the scenario solution is feasible for the original chance-constrained optimisation problem with confidence at least $1 - \beta$.

The proof of Theorem 3.1 is available in [117]. To explain the theorem in the context of Problems (3.22 - 3.23), let the solution of Problem (3.23) be \mathbf{u}^* (which is obtained from

the optimised variables, say Θ^* and Γ^*). If the number of scenarios N_r used in Problem (3.23) satisfies Eq. 3.24, then the following holds true

$$\mathbb{P}^{N_r} \{ \mathbb{P} \{ u_{\min} \leq \|\mathbf{u}^*\|_{\infty} \leq u_{\max} \cap \Delta u_{\min} \leq \|\Delta \mathbf{u}^*\|_{\infty} \leq \Delta u_{\max} \\ \cap y_{\min} \leq \|\mathbf{C}\mathbf{x}_n(\mathbf{u}^*)\|_{\infty} \leq y_{\max} \} \geq 1 - \epsilon \} \geq 1 - \beta,$$

where \mathbb{P}^{N_r} denotes a probability measure on the N_r extracted samples of \mathbf{w}_n and β is a confidence parameter. The solution \mathbf{u}^* is stochastic, because it depends on the N_r drawn scenarios of \mathbf{w}_n in Problem (3.23).

The parameter β can be explained as follows: we cannot guarantee that the scenario solution is always feasible for Problem (3.22), because it might happen that the N_r extracted realisations are not representative enough. However, if we meet the criterion in Eq. 3.24, then the probability of such an event is less than β , and the feasibility of the scenario solution is ensured with a confidence $1 - \beta$. In [118], it is shown that Eq. 3.24 holds true whenever N_r satisfies the explicit expression,

$$N_r \geq \frac{d + 1 + \ln(1/\beta) + \sqrt{2(d + 1)\ln(1/\beta)}}{\epsilon}. \quad (3.25)$$

The above expression explains the relationship between N_r and β , where N_r depends logarithmically on β , and hence β can be chosen very small (e.g. 10^{-6}) without increasing N_r too much. The scenario-based optimisation problems can be solved by standard convex optimisation solvers as e.g. used by YALMIP [3], CVX [119] etc.

3.3 State space representation of rivers

In this section we express the river model structures from Chapter 2 as state space models, to be used in the S-MPC setting developed in Section 3.1.

3.3.1 State space model for a general river reach

In Chapter 2, a river is modelled as a cascade of time delays and integrators, where each river reach is modelled as a transport delay and each storage along the river, e.g. a lake

or a dam is modelled as an integrator. In this chapter we consider a general river, and based on Eq. 2.5 we propose the following model structure for a general river reach,

$$\begin{aligned}
y_j(n+1) = y_j(n) &+ T \sum_{i=1}^{n_{cs}} c_{cs,i} Q_{cs,i}(n - \tau_{cs,i}) + T \sum_{i=1}^{n_{us}} c_{us,i} Q_{us,i}(n - \tau_{us,i}) \\
&+ T \sum_{i=1}^{n_{cr}} c_{cr,i} Q_{cr,i}(n - \tau_{cr,i}) + T \sum_{i=1}^{n_{ur}} c_{ur,i} Q_{ur,i}(n - \tau_{ur,i}), \quad (3.26)
\end{aligned}$$

where $j = 1, 2, \dots, n_{st}$, n_{st} is the total number of storages in the river and y_j is the water level in the j^{th} storage. To simplify notation, we have omitted the subscript 'j' from the flow variables and parameters. n_{cs} and n_{cr} are the number of regulated in- and out-flows, and n_{us} and n_{ur} are the number of unregulated in- and out-flows. Q_{cs} and Q_{cr} are the regulated in- and out-flows. Similarly, Q_{us} and Q_{ur} are the unregulated in- and out-flows. The flows are divided into regulated or unregulated groups because, for control problems, we need to distinguish between the flows that we can manipulate (the regulated flows) and those which we cannot (the unregulated flows). The regulated flows are mostly measured, however, the unregulated flows can be measured or unmeasured, and based on the availability of measurements, forecast models of the unregulated flows may be required for control purposes. τ , in Eq. 3.26, represents time delay from a source or sink to the storage, and T is the sampling interval.

For simplicity, we consider $n_{st} = n_{cs} = n_{us} = n_{cr} = n_{ur} = 1$, and construct a state space model from Eq. 3.26. The extension to the general case is straightforward. The state space model is constructed below by defining the states as delayed in- and out-flows, with the first state as the water level in the storage. For compact notations, time indices are given in subscripts,

$$\begin{aligned}
x_{1,n+1} &= x_{1,n} + c_{cs}x_{2,n} + c_{us}x_{\tau_{cs}+2,n} + c_{cr}x_{\tau_{cs}+\tau_{us}+2,n} + c_{ur}x_{\tau_{cs}+\tau_{us}+\tau_{cr}+2,n}, \\
x_{i,n+1} &= x_{i+1,n}, \quad \text{for } i = 2, \dots, \tau_{cs}, \\
x_{\tau_{cs}+1,n+1} &= Q_{cs,n}, \\
x_{i,n+1} &= x_{i+1,n}, \quad \text{for } i = \tau_{cs} + 2, \dots, \tau_{cs} + \tau_{us}, \\
x_{\tau_{cs}+\tau_{us}+1,n+1} &= Q_{us,n},
\end{aligned}$$

$$\begin{aligned}
x_{i,n+1} &= x_{i+1,n}, & \text{for } i &= \tau_{cs} + \tau_{us} + 2, \dots, \tau_{cs} + \tau_{us} + \tau_{cr}, \\
x_{\tau_{cs}+\tau_{us}+\tau_{cr}+1,n+1} &= Q_{cr,n}, \\
x_{i,n+1} &= x_{i+1,n}, & \text{for } i &= \tau_{cs} + \tau_{us} + \tau_{cr} + 2, \dots, \tau_{cs} + \tau_{us} + \tau_{cr} + \tau_{ur}, \\
x_{\tau_{cs}+\tau_{us}+\tau_{cr}+\tau_{ur}+1,n+1} &= Q_{ur,n},
\end{aligned}$$

and thus the state vector is given as,

$$\begin{aligned}
x_n &= [x_{1,n} \ x_{2,n} \ \dots \ x_{\tau_{cs}+1,n} \ x_{\tau_{cs}+2,n} \ \dots \ x_{\tau_{cs}+\tau_{us}+1,n} \ x_{\tau_{cs}+\tau_{us}+2,n} \ \dots \\
&\quad x_{\tau_{cs}+\tau_{us}+\tau_{cr}+1,n} \ x_{\tau_{cs}+\tau_{us}+\tau_{cr}+2,n} \ \dots \ x_{\tau_{cs}+\tau_{us}+\tau_{cr}+\tau_{ur}+1,n}]^\top \\
&= [y_{j,n} \ Q_{cs,n-\tau_{cs}} \ \dots \ Q_{cs,n-1} \ Q_{us,n-\tau_{us}} \ \dots \ Q_{us,n-1} \ Q_{cr,n-\tau_{cr}} \ \dots \\
&\quad Q_{cr,n-1} \ Q_{ur,n-\tau_{ur}} \ \dots \ Q_{ur,n-1}]^\top, \tag{3.27}
\end{aligned}$$

and the input vector is $u_n = [Q_{cs,n} \ Q_{us,n} \ Q_{cr,n} \ Q_{ur,n}]^\top$. We get

$$x_{n+1} = Ax_n + Bu_n, \tag{3.28}$$

as a state space model of Eq. 3.26, where

$$A = \begin{bmatrix}
1 & c_{cs} & \mathbf{0}'_{\tau_{cs}} & c_{us} & \mathbf{0}'_{\tau_{us}} & c_{cr} & \mathbf{0}'_{\tau_{cr}} & c_{ur} & \mathbf{0}'_{\tau_{ur}} \\
\mathbf{0}'_{\tau_{cs}} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
0 & 0 & \mathbf{0} & 0 & \mathbf{0} & 0 & \mathbf{0} & 0 & \mathbf{0} \\
\mathbf{0}'_{\tau_{us}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
0 & 0 & \mathbf{0} & 0 & \mathbf{0} & 0 & \mathbf{0} & 0 & \mathbf{0} \\
\mathbf{0}'_{\tau_{cr}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\
0 & 0 & \mathbf{0} & 0 & \mathbf{0} & 0 & \mathbf{0} & 0 & \mathbf{0} \\
\mathbf{0}'_{\tau_{ur}} & \mathbf{0} & \mathbf{I} \\
0 & 0 & \mathbf{0} & 0 & \mathbf{0} & 0 & \mathbf{0} & 0 & \mathbf{0}
\end{bmatrix}, \quad B = \begin{bmatrix}
0 & 0 & 0 & 0 \\
\mathbf{0}'_{\tau_{cs}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
1 & 0 & 0 & 0 \\
\mathbf{0}'_{\tau_{us}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
0 & 1 & 0 & 0 \\
\mathbf{0}'_{\tau_{cr}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
0 & 0 & 1 & 0 \\
\mathbf{0}'_{\tau_{ur}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
0 & 0 & 0 & 1
\end{bmatrix},$$

where $\tau'_x = \tau_x - 1$ and the bold elements indicate a vector or a matrix. The dimensions of the matrix blocks are indicated in the first row and column of the A matrix, and in the first column of the B matrix.

As mentioned above, we distinguish between the regulated and unregulated flows, so we rearrange and partition the columns in B (and the corresponding flows) as $B = \begin{bmatrix} B_C & B_U \end{bmatrix}$, leading to the following model,

$$x_{n+1} = Ax_n + B_C u_{C,n} + B_U u_{U,n}, \quad (3.29)$$

where $u_{C,n}$ and $u_{U,n}$ contain the regulated and unregulated flows respectively. In the general case, the number of columns of matrices B_C and B_U are equal to $n_{cs} + n_{cr}$ and $n_{us} + n_{ur}$ respectively.

River operators use forecasts of the unregulated in- and out-flows to make decisions. The forecasts are uncertain since they depend on weather prediction, upstream or downstream flows and a few other factors. The unregulated flows are therefore modelled as

$$u_{U,n} = u_{U,n}^f + w_n, \quad (3.30)$$

where $u_{U,n}^f$ is a forecast (f) of the unregulated flows at time n and w_n is a stochastic random variable used to model the uncertainty, which reflects the quality of the forecast. Forecast models are briefly discussed in the next subsection.

Substituting Eq. 3.30 in Eq. 3.29 gives

$$x_{n+1} = Ax_n + B_C u_{C,n} + B_U (u_{U,n}^f + w_n). \quad (3.31)$$

Finally let

$$w_{U,n} = B_U (u_{U,n}^f + w_n), \quad (3.32)$$

and Eq. 3.31 becomes

$$x_{n+1} = Ax_n + B_C u_{C,n} + w_{U,n}, \quad (3.33)$$

Instead of rearranging and partitioning the columns in matrix B (and the corresponding flows) in Eq. 3.28, into just regulated and unregulated categories, as in Eq. 3.31, we can use an alternative and elaborated description, where we partition the B matrix

into four categories. The categories are based on the associated flows, which can be regulated, unregulated-measured, unregulated-unmeasured-forecasted, and unregulated-unmeasured-unforecasted. Such a partition will lead to the following model,

$$x_{n+1} = Ax_n + B_C u_{C,n} + B_U^m u_{U,n}^m + B_U^f (u_{U,n}^f + w_n^f) + B_U^X u_{U,n}^X, \quad (3.34)$$

where $u_{U,n}^m$ is a measurement (m) of the unregulated flows that can be measured at time n , $u_{U,n}^f$ is a forecast (f) of the unregulated flows that are not measured at time n , w_n is a stochastic random variable used to model the uncertainty in the forecast and $u_{U,n}^X$ is a vector of unregulated flows that are neither measured nor forecasted, and B_U^m , B_U^f and B_U^X are the corresponding matrices that further partition the columns of B_U matrix in Eq. 3.31. With this formulation and by letting

$$w_{U,n} = B_U^f (u_{U,n}^f + w_n^f) + B_U^X u_{U,n}^X, \quad (3.35)$$

we get the equivalent river description as,

$$x_{n+1} = Ax_n + B_C u_{C,n} + B_U^m u_{U,n}^m + w_{U,n}. \quad (3.36)$$

However, we do not pursue this formulation in this thesis, because mostly unregulated flows are either measured or forecasted. Also, usually forecasts of most of the unregulated flows are required due to the time delays in the system, which cause the unavailability of the relevant measurements at the required time. So we assume only two categories to partition B matrix in this work: regulated and unregulated, where we assume the unregulated flows are forecasted, and Eqs. 3.31, 3.32 & 3.33 provide the formulation for all such cases. Alternatively, formulation with Eqs. 3.34, 3.35 & 3.36 can also be pursued ahead with minor modifications.

Eq. 3.33 has the same form as Eq. 3.5, and a description over a finite horizon M , as in Eq. 3.9, is given by,

$$\mathbf{x}_{n+1} = Fx_n + G\mathbf{u}_{C,n} + H\mathbf{w}_{U,n}, \quad (3.37)$$

where $\mathbf{x}_{n+1} = \begin{bmatrix} x_{n+1} & x_{n+2} & \dots & x_{n+M} \end{bmatrix}^\top$, $\mathbf{u}_{C,n} = \begin{bmatrix} u_{C,n} & u_{C,n+1} & \dots & u_{C,n+M-1} \end{bmatrix}^\top$ and $\mathbf{w}_{U,n} = \begin{bmatrix} w_{U,n} & w_{U,n+1} & \dots & w_{U,n+M-1} \end{bmatrix}^\top$ respectively. The matrices F , G and H are given by

$$F = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^M \end{bmatrix}, \quad G = \begin{bmatrix} B_C & \mathbf{0} & \dots & \mathbf{0} \\ AB_C & B_C & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ A^{M-1}B_C & A^{M-2}B_C & \dots & B_C \end{bmatrix}, \quad H = \begin{bmatrix} I & \mathbf{0} & \dots & \mathbf{0} \\ A & I & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ A^{M-1} & A^{M-2} & \dots & I \end{bmatrix}.$$

Furthermore, for S-MPC based control for rivers, we use the control policy given in Eq. 3.16, i.e.

$$\mathbf{u}_{C,n} = \Theta_n \mathbf{w}_{U,n} + \Gamma_n, \quad (3.38)$$

and we select the parametrisation of Γ_n and Θ_n matrices given in Eq. 3.19. To use Eq. 3.38 as a control policy, we need the value of $w_{U,n-1}$ in $\mathbf{w}_{U,n}$ matrix, at time n , which can be obtained from the state measurements at time n using Eq. 3.33 as,

$$w_{U,n-1} = x_n - Ax_{n-1} - B_C u_{C,n-1}.$$

The above expression is computable, only if the measurement of the full state vector, x_n , in Eq. 3.27 is available. We assume the measurements are available in this work, because in rivers, the control variables (e.g. water level in a lake) and the variables which affect the control variables (regulated and unregulated in- and out-flows) are usually measured.

3.3.2 Unregulated flow forecast

As described earlier, we need to distinguish between the flows that can and cannot be regulated for control purposes. The unregulated flows act like disturbances and their forecasts are usually required, e.g. they are needed when deciding the amount of water that should be released from storages to meet farmers and environmental demands and/or to avoid flooding. We built flow forecast models of unregulated rivers in Appendices 2.C & 2.D. In this section we briefly describe some basic model structures for

unregulated flow forecasts and their uncertainties.

In Eq. 3.32, $w_{U,n}^f = \begin{bmatrix} Q_{us,n}^f & Q_{ur,n}^f \end{bmatrix}^\top$ is a vector of forecasts of unregulated flows. An unregulated inflow $Q_{us,n}$ can be forecast as a weighted sum of its upstream flows, provided their measurements are available i.e.

$$Q_{us,n}^f = \phi_{us}^1 Q_{us,n-\tau_{us}^1}^1 + \phi_{us}^2 Q_{us,n-\tau_{us}^2}^2 + \dots + \phi_{us}^{n_{uus}} Q_{us,n-\tau_{us}^{n_{uus}}}^{n_{uus}}, \quad (3.39)$$

where $Q_{us,n-\tau_{us}^i}^i$ is the i^{th} upstream flow contributing to the unregulated inflow at time $n - \tau_{us}^i$, with $i = 1, \dots, n_{uus}$, and n_{uus} is the number of upstream flows contributing to the unregulated inflow $Q_{us,n}$. τ_{us}^i is the time delay from the i^{th} upstream flow, and ϕ_{us}^i is a parameter associated with the i^{th} upstream flow. Rainfall and weather forecasts can also be incorporated in the flow forecast, but they are not considered here.

An unregulated outflow $Q_{ur,n}$ is usually a diversion to an irrigation channel. However, in general, it includes all uncertain demands, e.g. demands from farmers (f), environment (e) and some others (r), which could be small water supplies etc. For n_{ur} unregulated outflows we can model $Q_{ur,n}^f$ as,

$$Q_{ur,n}^f = \psi_{ur}^f D_{ur,n}^f + \psi_{ur}^e D_{ur,n}^e + \psi_{ur}^r D_{ur,n}^r, \quad (3.40)$$

where $D_{ur,n}$ is a flow demand at time n and ψ_{ur}^f , ψ_{ur}^e and ψ_{ur}^r are the parameters associated with the demands, which can be estimated from past operational data.

The vector $w_n = \begin{bmatrix} w_{us,n} & w_{ur,n} \end{bmatrix}^\top$ in Eq. 3.32 represents the uncertainty in the forecasts. They can be modelled as independent disturbances, or disturbances correlated in time and (or) space e.g. by using an Auto-Regressive (AR) process (used in [120]). In the latter case, the assumption of independent and identically distributed disturbance will not remain valid. However, an AR process with temporal correlation can naturally describe some of the uncertainty, e.g. it is more likely to be a rainy or sunny day tomorrow, if it is rainy or sunny today. Parameters in an AR model can also be estimated from historical data.

Flow forecasting is not a part of this thesis, and the above material is included just to illustrate that various types of flow forecasts can be easily incorporated within the proposed control formulation.

In Appendix 3.A we discuss two additional topics on feedback control policies that can be pursued in river control. We discuss (i) a control policy that can use forecasts of

the disturbance term $w_{U,n}$, (ii) how to incorporate an integral action in the developed formulation.

3.4 Stochastic MPC for rivers

In this section we discuss common constraints and objectives in river operations and state them mathematically, before we use them to formulate a river optimisation problem that can be solved in an S-MPC setting.

3.4.1 Common constraints in rivers

A typical river observes the following restrictions due to physical and environmental constraints.

- In water storages, flow regulation is usually available at the downstream end of the storages by means of hydraulic structures. The flow regulation is used for general management of the storage and other water resources, downstream of the storage. The excessive flow release is restricted to avoid flooding, and a minimum release is usually observed for environmental purposes.
- Diversions to irrigation channel or wetlands are mostly regulated, because the water is mostly diverted only to meet farmers or environmental demands.
- Rapid changes in river flows should be avoided to preserve the river bank and avoid river bank slumping, where the mud beneath the river bank is forcefully dragged along with the water flow.

Next, we state these constraints mathematically. For the regulated inflows we have the following constraints

$$Q_{cs,i}^{\min} \leq Q_{cs,i,n} \leq Q_{cs,i}^{\max}, \quad (3.41)$$

$$\Delta Q_{cs,i}^{\min} \leq Q_{cs,i,n} - Q_{cs,i,n-1} \leq \Delta Q_{cs,i}^{\max}, \quad (3.42)$$

for all $i = 1, 2, \dots, n_{cs}$ at each time step n , i.e. all regulated inflows and their change in time are restricted between upper and lower limits. Moreover, the upper and lower limits can be time varying as well. The regulated outflows impose similar constraints,

$$Q_{cr,i}^{\min} \leq Q_{cr,i,n} \leq Q_{cr,i}^{\max}, \quad (3.43)$$

$$\Delta Q_{cr,i}^{\min} \leq Q_{cr,i,n} - Q_{cr,i,n-1} \leq \Delta Q_{cr,i}^{\max}, \quad (3.44)$$

for all $i = 1, 2, \dots, n_{cr}$ at each time step n , i.e. all regulated outflows and their change in time are restricted between upper and lower limits. Similarly, water level in a storage is also maintained within limits i.e.

$$y_i^{\min} \leq y_{i,n} \leq y_i^{\max}, \quad (3.45)$$

for all $i = 1, 2, \dots, n_{st}$ at each time step n .

As explained earlier in Section 3.2.3, a hard constraint on the output (water level) is not advisable as the disturbance $w_{U,n}$ (Eq. 3.33) is additive, and we assume an unbounded support on the disturbance. To compensate for that, we consider 'soft and soft' category of constraints in which the constraints on the output and control inputs are both kept soft (probabilistic), e.g. see Eq. 3.21. Next, we express the above constraints (Eqs. 3.41, 3.42, 3.43, 3.44 & 3.45) probabilistically, in a compact way, for the finite horizon M . We partition $u_{C,n}$ in Eq. 3.29 as,

$$u_{C,n} = \begin{bmatrix} Q_{cs,i,n} \\ Q_{cr,j,n} \end{bmatrix}_{\substack{i=1, j=1 \\ i=n_{cs}, j=n_{cr}}}.$$

Thus, for a finite horizon M , the vector $\mathbf{u}_{C,n} = [u_{C,n} \quad u_{C,n+1} \quad \dots \quad u_{C,n+M-1}]^T$ in Eq. 3.37 can be constrained as,

$$\mathbf{u}_{\min} \leq \mathbf{u}_{C,n} \leq \mathbf{u}_{\max}, \quad (3.46)$$

where

$$\mathbf{u}_{\min} = \begin{bmatrix} Q_{cs,i}^{\min} \\ Q_{cr,i}^{\min} \end{bmatrix}_{i=1,j=1}^{i=n_{cs},j=n_{cr}}, \quad \mathbf{u}_{\max} = \begin{bmatrix} Q_{cs,i}^{\max} \\ Q_{cr,i}^{\max} \end{bmatrix}_{i=1,j=1}^{i=n_{cs},j=n_{cr}},$$

and the inequality ' \leq ' in Eq. 3.46 is interpreted element by element. Similarly, Eqs. 3.42 & 3.44 can be written as,

$$\Delta \mathbf{u}_{\min} \leq \mathbf{u}_{C,n} - \mathbf{u}_{C,n-1} \leq \Delta \mathbf{u}_{\max}, \quad (3.47)$$

where

$$\Delta \mathbf{u}_{\min} = \begin{bmatrix} \Delta Q_{cs,i}^{\min} \\ \Delta Q_{cr,i}^{\min} \end{bmatrix}_{i=1,j=1}^{i=n_{cs},j=n_{cr}}, \quad \Delta \mathbf{u}_{\max} = \begin{bmatrix} \Delta Q_{cs,i}^{\max} \\ \Delta Q_{cr,i}^{\max} \end{bmatrix}_{i=1,j=1}^{i=n_{cs},j=n_{cr}}.$$

The probabilistic constraints on control policies (as in Eq. 3.21) can be collectively given from Eqs. 3.46 & 3.47 as below,

$$\mathbb{P}\{\mathbf{w}_{U,n} \in \mathbb{W} : \mathbf{u}_{\min} \leq \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \mathbf{u}_{\max} \cap \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \Delta \mathbf{u}_{\max}\} \geq 1 - \epsilon, \quad (3.48)$$

where \mathbb{W} is a convex set to which the disturbances $\mathbf{w}_{U,n}$ belong, \mathbb{P} is a probability measure on the set \mathbb{W} , $\Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) = \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) - \mathbf{u}_{C,n-1}(\mathbf{w}_{U,n-1})$ and $\epsilon \in (0,1)$ is a user chosen probability level.

Similarly, to write Eq. 3.45 probabilistically, in a compact way, for the finite horizon M (for a single storage, i.e. $i = 1$ in Eq. 3.45), consider

$$y_{\text{ref}} = \frac{y^{\min} + y^{\max}}{2}, \quad (3.49)$$

and $y_m = y^{\max} - y_{\text{ref}} = y_{\text{ref}} - y^{\min}$. y_{ref} is going to be used in the water level constraints, rather than y^{\min} and y^{\max} , because it helps in making the formulation of the optimisation schemes (in Section 3.5) easier.

The constraints on the output can be written as

$$y_{\text{ref}} - y_m \leq \|\mathbf{C}\mathbf{x}_n\|_{\infty} \leq y_{\text{ref}} + y_m, \quad (3.50)$$

where the infinity norm gives the maximum absolute value of the element in the vector, \mathbf{x}_n is given by Eq. 3.37 and $\mathbf{C} = \text{diag}(C, \dots, C)$, and for a single storage the matrix C has the first element equal to 1 with rest equal to zero. The probabilistic constraint on the states (as in Eq. 3.21) can be given as

$$\mathbb{P}\{\mathbf{w}_{U,n} \in \mathbb{W} : y_{\text{ref}} - y_m \leq \|\mathbf{C}\mathbf{x}_n(\mathbf{w}_{U,n})\|_{\infty} \leq y_{\text{ref}} + y_m\} \geq 1 - \epsilon. \quad (3.51)$$

Eqs. 3.48 & 3.51 cover most of the physical and environmental constraints associated with rivers, and we use them to define a river optimisation problem in the next sub-section.

3.4.2 Objective function

In this section, we use the objective function considered in Eq. 3.7 (or in Problem (3.23)) and present the river optimisation problem. The objective function in Problem (3.23) accommodates most of the control objectives of a typical river, which include minimisation of

- Deviation of states from their reference values, e.g. minimisation of the deviation of water levels from their set points. Water levels in lakes along the river are controlled, and they are required to be maintained between upper and lower limits (Eq. 3.45), determined by river operators. A reference set point can be obtained by taking the mean of these limits (Eq. 3.49).
- Variation in control inputs, i.e. to reduce changes in flow release. This is important to preserve the river bank structure.
- Control inputs themselves, i.e. to minimise flow release from storages to avoid waste of water.

It is therefore natural to consider the following objective function for the river control problem,

$$J(\Theta_n, \Gamma_n) = \mathbb{E}[(\mathbf{x}_{n+1} - \mathbf{x}_r)^\top \mathbf{Q}(\mathbf{x}_{n+1} - \mathbf{x}_r) + \mathbf{u}_{C,n}^\top \mathbf{R}\mathbf{u}_{C,n} + (\mathbf{u}_{C,n} - \mathbf{u}_{C,n-1})^\top \mathbf{S}(\mathbf{u}_{C,n} - \mathbf{u}_{C,n-1})], \quad (3.52)$$

where \mathbb{E} is the conditional expectation over the disturbance term $\mathbf{w}_{U,n}$ given the initial state x_n . \mathbf{x}_r is a vector of reference state values which contain the reference water level, and the matrices \mathbf{Q} , \mathbf{R} and \mathbf{S} are described earlier, in Eq. 3.7 and Problem (3.23)).

Next, we accommodate the expressions of the states \mathbf{x}_{n+1} and the control policy $\mathbf{u}_{C,n}$ from Eqs. 3.37 and 3.38 in the objective function (Eq. 3.52), and obtain

$$\begin{aligned}
J(\Theta_n, \Gamma_n) &= (Fx_n + G\Gamma_n - \mathbf{x}_r)^\top \mathbf{Q}(Fx_n + G\Gamma_n - \mathbf{x}_r) + 2(Fx_n + G\Gamma_n - \mathbf{x}_r)^\top \mathbf{Q}(H \\
&+ G\Theta_n) \cdot \mathbb{E}[\mathbf{w}_{U,n}] + \text{tr}[(H + G\Theta_n)^\top \mathbf{Q}(H + G\Theta_n) \cdot \mathbb{E}[\mathbf{w}_{U,n}\mathbf{w}_{U,n}^\top]] \\
&+ (\Gamma_n - \Gamma_{n-1})^\top \mathbf{R}(\Gamma_n - \Gamma_{n-1}) + 2(\Gamma_n - \Gamma_{n-1})^\top \mathbf{R}(\Theta_n - \Theta_{n-1}) \\
&\cdot \mathbb{E}[\mathbf{w}_{U,n}] + \text{tr}[(\Theta_n - \Theta_{n-1})^\top \mathbf{R}(\Theta_n - \Theta_{n-1}) \cdot \mathbb{E}[\mathbf{w}_{U,n}\mathbf{w}_{U,n}^\top]] \\
&+ \Gamma_n^\top \mathbf{S}\Gamma_n + 2\Gamma_n^\top \mathbf{S}\Theta_n \cdot \mathbb{E}[\mathbf{w}_{U,n}] + \text{tr}[\Theta_n^\top \mathbf{S}\Theta_n \cdot \mathbb{E}[\mathbf{w}_{U,n}\mathbf{w}_{U,n}^\top]],
\end{aligned} \tag{3.53}$$

where, corresponding to the parametrisation in Eq. 3.17, Γ_{n-1} and Θ_{n-1} are given as

$$\Gamma_{n-1} = \underbrace{\begin{bmatrix} u_{C,n-1}^* \\ \gamma_n \\ \gamma_{n+1} \\ \vdots \\ \gamma_{n+M-2} \end{bmatrix}}_{(Mm \times 1)}, \quad \Theta_{n-1} = \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \theta_{n+1,0} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \theta_{n+M-2,0} & \theta_{n+M-2,1} & \dots & \mathbf{0} \end{bmatrix}}_{(Mm \times Mn_s)}, \tag{3.54}$$

where $u_{C,n-1}^*$ is the control action applied at time $n-1$, and from Eq. 3.32 we have

$$\mathbb{E}[\mathbf{w}_{U,n}] = \begin{bmatrix} B_U(u_{U,n}^f + \mathbb{E}[w_n]) \\ \vdots \\ B_U(u_{U,n+M-1}^f + \mathbb{E}[w_{n+M-1}]) \end{bmatrix},$$

where $\mathbb{E}[w_n]$ is the mean value of disturbance w at time n . Similarly, $\mathbb{E}[\mathbf{w}_{U,n}\mathbf{w}_{U,n}^\top]$ can be computed by taking expectation of the product of $\mathbf{w}_{U,n}$ with its transpose.

Finally, using the river constraints (Eqs. 3.48 & 3.51) and the objective function (Eq. 3.53), we define the river optimisation problem at time n as the following chance-constrained optimisation problem (CCP),

$$\min_{\Theta_n, \Gamma_n} J(\Theta_n, \Gamma_n), \tag{3.55}$$

$$\text{s.t. } \mathbb{P}\{\mathbf{w}_{U,n} \in \mathbb{W} : \mathbf{u}_{\min} \leq \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \mathbf{u}_{\max} \cap \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \Delta \mathbf{u}_{\max}$$

$$\cap y_{\text{ref}} - y_m \leq \|\mathbf{C}\mathbf{x}_n(\mathbf{w}_{U,n})\|_\infty \leq y_{\text{ref}} + y_m\} \geq 1 - \epsilon,$$

where the system dynamics \mathbf{x}_n is given in Eq. 3.37 and the control policy $\mathbf{u}_{C,n}$ is given in Eq. 3.38.

We aim to solve Problem (3.55) in an MPC set-up. However, as mentioned earlier a CCP is difficult to solve, as a probabilistic constraints are generally non-convex with respect to optimisation variables. To find an approximate solution to Problem (3.55), we employ the scenario approach (Section 3.2.4) and we get the following scenario version of Problem (3.55), which is convex with respect to the optimisation variables (Θ_n, Γ_n) ,

$$\begin{aligned} & \min_{\Theta_n, \Gamma_n} J(\Theta_n, \Gamma_n), & (3.56) \\ \text{s.t. } & \mathbf{u}_{\min} \leq \mathbf{u}_{C,n}(\mathbf{w}_{U,n}^{(k)}) \leq \mathbf{u}_{\max}, \quad \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}^{(k)}) \leq \Delta \mathbf{u}_{\max} \\ & y_{\text{ref}} - y_m \leq \|\mathbf{C}\mathbf{x}_n(\mathbf{w}_{U,n}^{(k)})\|_{\infty} \leq y_{\text{ref}} + y_m, \quad \text{for } k = 1, \dots, N_r, \end{aligned}$$

where N_r is the number of the disturbance ($\mathbf{w}_{U,n}$) realisations, which can be obtained from Eq. 3.24 by using ϵ in Problem (3.55) and selecting a very small confidence parameter β (e.g. 10^{-6}). Problem (3.56) is a convex optimisation problem, and it can be solved by standard convex optimisation solvers as e.g. used by YALMIP [3], CVX [119] etc.

Even with the convexity assurance, the feasibility of the scenario Problem (3.56) is still not guaranteed, because of the presence of the unbounded disturbance terms $\mathbf{w}_{U,n}^{(k)}$ in the constraints, where $k = 1, 2, \dots, N_r$. The feasibility issues can be critical, especially when y_m is a small number. In the next section we introduce two optimisation schemes for the river control problem that ensure feasibility, and we can apply the scenario approach to solve them.

3.5 Stochastic MPC schemes for rivers with feasibility assurance

In this section we consider the river control Problem (3.55), and formulate two alternative chance-constrained optimisation schemes that ensure feasibility and which can be solved using the scenario approach (Section 3.2.4).

As a general idea, we replace y_m in Eq. 3.51 with a decision variable, h , to ensure feasibility. However, we have to specify an appropriate trade-off between performance,

as measured by the value of J in Eq. 3.52, and feasibility. In this section we formulate two MPC schemes based on this idea. For river problems, such an arrangement is acceptable as water level often crosses the lower and upper limits ($y_{\text{ref}} - y_m$ and $y_{\text{ref}} + y_m$), and setting y_m as a decision variable h will allow room for that, however, the optimisation schemes will penalise such events. Scheme 1 is borrowed from [103], which is based on a user chosen trade-off between performance and feasibility. Scheme 2 is an automated option which does not require any user input.

Scheme 1 (a two-step approach with a tuning parameter)

In this scheme two optimisation problems are solved. In the first problem the objective function J is minimised subject to the constraints on the input only. In the second problem constraint satisfaction is improved by minimising the bound h , subject to constraints on both the input and the states, and a constraint that defines the maximum allowed degradation of the objective value achieved in the previous problem by means of a tuning parameter α .

Problem 1

$$\min_{\Theta_n, \Gamma_n} J(\Theta_n, \Gamma_n), \quad (3.57)$$

$$\text{s.t. } \mathbb{P}\{\mathbf{w}_{U,n} \in \mathbb{W} : \mathbf{u}_{\min} \leq \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \mathbf{u}_{\max} \cap \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \Delta \mathbf{u}_{\max}\} \geq 1 - \epsilon,$$

subject to the dynamics given by Eqs. 3.37 & 3.38. Suppose J^* is the optimal value achieved by solving this problem

Problem 2

$$\min_{\Theta_n, \Gamma_n, h} h, \quad (3.58)$$

$$\text{s.t. } \mathbb{P}\{\mathbf{w}_{U,n} \in \mathbb{W} : \mathbf{u}_{\min} \leq \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \mathbf{u}_{\max} \cap \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \Delta \mathbf{u}_{\max}$$

$$\cap y_{\text{ref}} - h \leq \|\mathbf{C}\mathbf{x}_n(\mathbf{w}_{U,n})\|_{\infty} \leq y_{\text{ref}} + h\} \geq 1 - \epsilon,$$

$$J(\Theta_n, \Gamma_n) \leq J^*(1 + \alpha),$$

where $\alpha \geq 0$ is a tuning parameter that determines the trade-off between performance (J) and feasibility (h). In this scheme we can define a percentage degradation of the objective value by selecting $\alpha = 10\%$, 20% etc.

Scheme 2 (an automated two-step approach)

This scheme also consists of two optimisation problems. However, it does not require selection of any user chosen parameters. In the first problem, we ignore the objective function J and minimise h subject to the chance constraints on states and input. Then we check the program feasibility by comparing the achieved bound with a strict bound, $\bar{h} = y_m$ (as in Eq. 3.50). If the program is feasible, then the objective function J is minimised in the second optimisation problem subject to the original chance constraints. Otherwise, the solution of the first problem: $(\Theta^I, \Gamma^I, h^I)$ is used.

Problem 1

$$\begin{aligned} & \min_{\Theta_n, \Gamma_n, h} h, & (3.59) \\ \text{s.t. } & \mathbb{P}\{\mathbf{w}_{U,n} \in \mathbb{W} : \mathbf{u}_{\min} \leq \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \mathbf{u}_{\max} \cap \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \Delta \mathbf{u}_{\max} \\ & \cap y_{\text{ref}} - h \leq \|\mathbf{C}\mathbf{x}_n(\mathbf{w}_{U,n})\|_{\infty} \leq y_{\text{ref}} + h\} \geq 1 - \epsilon, \end{aligned}$$

subject to the dynamics given by Eqs. 3.9 & 3.16. If $h^I > \bar{h}$, return the solution $(\Theta^I, \Gamma^I, h^I)$, otherwise, proceed to the next problem where the performance is improved by minimising J .

Problem 2

$$\begin{aligned} & \min_{\Theta_n, \Gamma_n} J(\Theta_n, \Gamma_n), & (3.60) \\ \text{s.t. } & \mathbb{P}\{\mathbf{w}_{U,n} \in \mathbb{W} : \mathbf{u}_{\min} \leq \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \mathbf{u}_{\max} \cap \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \Delta \mathbf{u}_{\max} \\ & \cap y_{\text{ref}} - \bar{h} \leq \|\mathbf{C}\mathbf{x}_n(\mathbf{w}_{U,n})\|_{\infty} \leq y_{\text{ref}} + \bar{h}\} \geq 1 - \epsilon. \end{aligned}$$

In the second problem, the original bound \bar{h} is used. The solution $(\Theta^{\text{II}}, \Gamma^{\text{II}})$ is returned.

3.5.1 Scenario approach applied to the optimisation schemes

The scenario approach (Section 3.2.4) is applied to Schemes 1 and 2 below.

Scenario version of Scheme 1 (Problems (3.57 - 3.58))

The scenario version of Problem (3.57) can be obtained by replacing every constraint of the problem with N_r constraints, each corresponding to a realisation of the disturbance term $\mathbf{w}_{U,n}$ as below,

Problem 1

$$\begin{aligned} & \min_{\Theta_n, \Gamma_n} J(\Theta_n, \Gamma_n), & (3.61) \\ \text{s.t. } & \mathbf{u}_{\min} \leq \mathbf{u}_{C,n}(\mathbf{w}_{U,n}^{(k)}) \leq \mathbf{u}_{\max}, \quad \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}^{(k)}) \leq \Delta \mathbf{u}_{\max} \\ & \text{for } k = 1, \dots, N_r. \end{aligned}$$

Suppose J^* is the optimal value achieved by solving the above problem. The scenario version of Problem (3.58) is then given by,

Problem 2

$$\begin{aligned} & \min_{\Theta_n, \Gamma_n, h} h, & (3.62) \\ \text{s.t. } & \mathbf{u}_{\min} \leq \mathbf{u}_{C,n}(\mathbf{w}_{U,n}^{(k)}) \leq \mathbf{u}_{\max}, \quad \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}^{(k)}) \leq \Delta \mathbf{u}_{\max}, \\ & y_{\text{ref}} - h \leq \|\mathbf{C}\mathbf{x}_n(\mathbf{w}_{U,n}^{(k)})\|_{\infty} \leq y_{\text{ref}} + h, \quad J(\Theta_n, \Gamma_n) \leq J^*(1 + \alpha), \\ & \text{for } k = 1, \dots, N_r, \end{aligned}$$

where $\alpha \geq 0$ is the user chosen tuning parameter (selected in Problem (3.58)). Note that, we use the same noise realisations $\mathbf{w}_{U,n}^{(k)}$ in the two problems, where $k = 1, \dots, N_r$.

Scenario version of Scheme 2 (Problems (3.59 - 3.60))

Similarly, the scenario version of Problem (3.59) can be obtained by replacing every constraint of the problem with N_r constraints, each corresponding to a realisation of the

disturbance term $\mathbf{w}_{U,n}$ as below,

Problem 1

$$\begin{aligned} & \min_{\Theta_n, \Gamma_n, h} h, & (3.63) \\ \text{s.t. } & \mathbf{u}_{\min} \leq \mathbf{u}_{C,n}(\mathbf{w}_{U,n}^{(k)}) \leq \mathbf{u}_{\max}, \quad \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}^{(k)}) \leq \Delta \mathbf{u}_{\max} \\ & y_{\text{ref}} - h \leq \|\mathbf{C}\mathbf{x}_n(\mathbf{w}_{U,n}^{(k)})\|_{\infty} \leq y_{\text{ref}} + h, \quad \text{for } k = 1, \dots, N_r. \end{aligned}$$

If $(\Theta^I, \Gamma^I, h^I)$ is the solution of the above problem, and $h^I \geq \bar{h}$, where $\bar{h} = y_m$ (Eq. 3.50), then the solution is returned. Otherwise, the solution is improved by solving the following scenario version of Problem (3.60) and its solution is returned.

Problem 2

$$\begin{aligned} & \min_{\Theta_n, \Gamma_n} J(\Theta_n, \Gamma_n), & (3.64) \\ \text{s.t. } & \mathbf{u}_{\min} \leq \mathbf{u}_{C,n}(\mathbf{w}_{U,n}^{(k)}) \leq \mathbf{u}_{\max}, \quad \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}^{(k)}) \leq \Delta \mathbf{u}_{\max} \\ & y_{\text{ref}} - \bar{h} \leq \|\mathbf{C}\mathbf{x}_n(\mathbf{w}_{U,n}^{(k)})\|_{\infty} \leq y_{\text{ref}} + \bar{h}, \quad \text{for } k = 1, \dots, N_r. \end{aligned}$$

Note that, we use the same noise realisations $\mathbf{w}_U^{(k)}$ in the two problems, where $k = 1, \dots, N_r$.

3.5.2 Feasibility of the scenario bases schemes

The scenario based schemes are feasible by design, however, due to the cascaded nature, the confidence bound, β , on the scenario solution does not remain valid for the corresponding schemes with chance-constraints [103]. We discuss each scheme separately,

Feasibility of the scenario version of Scheme 1 (Problems (3.61 - 3.62))

In the scenario Problem (3.62) J^* depends on $\mathbf{w}_U^{(k)}$, since it is found by solving Problem (3.61), and J^* should be precisely written as $J^*(\mathbf{w}_U^{(1)}, \mathbf{w}_U^{(2)}, \dots, \mathbf{w}_U^{(N)})$. Hence there is a single constraint, ' $J(\Theta_n, \Gamma_n) \leq J^*(\mathbf{w}_U^{(1)}, \mathbf{w}_U^{(2)}, \dots, \mathbf{w}_U^{(N)})(1 + \alpha)$ ', in the scenario Problem (3.62), which depends on all of the scenarios, $\mathbf{w}_U^{(k)}$, and such situations are not covered

by Theorem 3.1. However, in practice, the solution is usually feasible for the original probabilistic constraints [103].

Feasibility of the scenario version of Scheme 2 (Problems (3.63 - 3.64))

The confidence bound β does not remain valid for Scheme 2 either, however, we have the following result based on Theorem 3.1.

Corollary 3.1. *Let $(\Theta^\dagger, \Gamma^\dagger, h^\dagger)$ be the solution of the cascade of Problems (3.63 - 3.64), which is assumed to always exist and be unique. For a user selected \bar{h} , and any $\epsilon \in (0, 1)$ and $\beta \in (0, 1)$, the probability of drawing scenarios $w_{U,n}^{(k)}$, where $k = 1, 2, \dots, N_r$ and N_r is as in Eq. 3.24, such that either $h^\dagger > \bar{h}$ and $(\Theta^\dagger, \Gamma^\dagger, h^\dagger)$ is infeasible for Problem (3.59) or $h^\dagger \leq \bar{h}$ and $(\Theta^\dagger, \Gamma^\dagger, h^\dagger)$ is infeasible for Problem (3.60), is at most 2β .*

A proof of Corollary 3.1 is given in Appendix 3.B, and according to the corollary, the solution of the cascade of Problems (3.63 - 3.64) is feasible for the cascade of Problems (3.59 - 3.60), with probability at least $1 - 2\beta$.

3.6 Application of the proposed schemes to the upper part of Murray River

In this section we apply the proposed scenario-based MPC schemes to the upper part of Murray River in Australia using historical data of the unregulated in- and out-flows. For ease, we re-present the sketch of the upper part of Murray River in Fig. 3.1. The figure shows the river stretch from Hume reservoir to Lake Mulwala. During normal operations the water level is controlled from Hume only. We briefly summarise the control objectives from Sections 1.3.2 & 2.1.1.

1. The water level in Lake Mulwala should be kept between 124.65 to 124.9 mAHD (meter Australian Height Datum—relative to sea level).
2. To avoid river bank slumping, the rate of fall in the flow release from Hume should be limited. The formal requirement is to keep the rate of fall in the water level

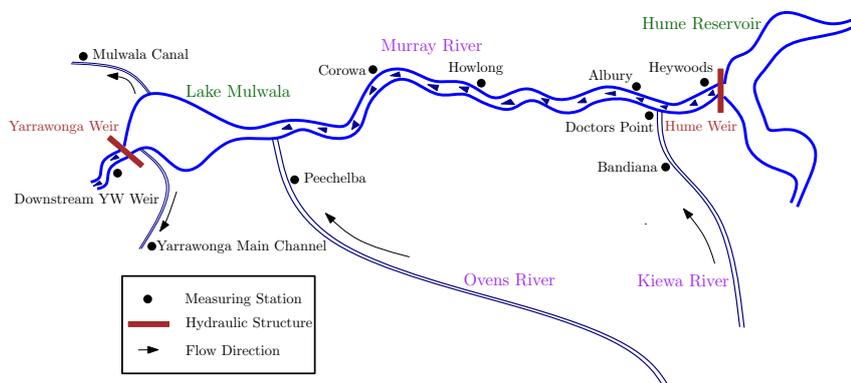


Figure 3.1: Hume Reservoir to Lake Mulwala on the Murray River (plot not to scale).

at Heywoods and Doctors Point below 0.20 m/Day and 0.15 m/Day respectively. However, in this work, we roughly approximate the requirement into the rate of fall in the flow at Heywoods, which should be below 500 ML/Day (where $1 \text{ m}^3/\text{sec} = 86.4 \text{ ML/Day}$). This approximation is obtained from historical data and the available rating curves at these locations. We found that this much flow variation at Heywoods keeps the rate of fall in the water level at Heywoods and Doctors Point below their allowed limits (most of the time).

3. The release from Hume Reservoir should be kept between 2,500 and 30,000 ML/Day.

For the MPC problem, we use the following model of water level in Lake Mulwala (obtained in Section 2.2.1),

$$y_{LM}(n+1) = y_{LM}(n) + 6.20 \times 10^{-7} Q_H(n-70) + 9.63 \times 10^{-7} Q_B(n-71) + 5.89 \times 10^{-7} Q_P(n-16) - 6.13 \times 10^{-7} Q_{DYW}(n) - 9.73 \times 10^{-7} Q_{YMC}(n) - 6.67 \times 10^{-7} Q_{MC}(n), \quad (3.65)$$

where Q_H , Q_B and Q_P are the inflows from Heywoods, Bandiana and Peechelba, and Q_{DYW} , Q_{YMC} and Q_{MC} are the releases to downstream of Yarrowonga Weir, Yarrowonga Main Channel and Mulwala Canal. As mentioned in Section 3.3.1, the MPC strategy uses river models in state space form, and in this section, we will use an equivalent state space model of Eq. 3.65 (shown in Appendix 3.C).

3.6.1 Control design

To test the performance of the proposed optimisation schemes, we used two different data with low and high inflows from the unregulated rivers (Kiewa and Ovens Rivers). To be precise, we picked one dataset from 7th Oct., 2006 to 20th Dec., 2006, and the other from 19th Sep., 2001 to 16th Nov., 2001, both sampled at $T_s = 8$ hours. We used a prediction horizon $M = 20$ units in the MPC problem, which is equal to 6.67 days (≈ 1 week). Based on the control objectives, we considered probabilistic versions of the following constraints,

- $124.65 \leq y_{i,LM} \leq 124.9$, for $i = 1, 2, \dots, M$,
- $2,500 \leq Q_{i,H} \leq 30,000$, for $i = 1, 2, \dots, M$,
- $-500 \leq Q_{i,H} - Q_{i-1,H} \leq 1,200$, for $i = 1, 2, \dots, M$.

The upper limit of the constraint on the change of flow at Heywoods is kept as 1,200 ML/Day, which is acceptable as the river operators put a constraint on the rate of fall in water level only. We have estimated the upper limit from historical data. The matrices \mathbf{Q} and \mathbf{S} in the objective function $J(\Theta_n, \Gamma_n)$ (Eq. 3.52) are selected as block diagonal matrices with M copies of the Q' and S' matrices on the block diagonals respectively, where

$$Q' = \begin{bmatrix} 10 & \mathbf{0}_{1, n_s-1} \\ \mathbf{0}_{n_s-1, 1} & \mathbf{0}_{n_s-1, n_s-1} \end{bmatrix} \quad S' = 10^{-11},$$

$n_s = 21$ is the number of states, and the first state corresponded to the water level in Lake Mulwala (Appendix 3.C). The large magnitude difference in the weights of Q' and S' matrices is due to large magnitude differences in the flow values and water levels at Heywoods and Lake Mulwala respectively. Q' and S' were tuned based on experiments on historical data. We used $\mathbf{R} = \mathbf{0}$ (in Eq. 3.52), since the change in flow at Heywoods is already subjected to constraints.

We used the parametrisation of Γ_n and Θ_n matrices (in Eq. 3.38) as in Eq. 3.19, to keep the number of optimisation variables, d , small. For the scenario MPC, we found that the number of scenarios must be $N_r \geq 1,250$ using Eq. 3.24, in order to meet the requirements:

$\epsilon = 0.1$ and $\beta = 1 \times 10^{-6}$, for $d = 78$. The requirements are set as follows. The violation probability ϵ is selected experimentally by observing violations at each MPC step and by keeping the corresponding computational burden affordable. The confidence parameter β is selected to be a very small number. The number of optimisation variables d is computed from Eq. 3.20 with $m = 6$ and $M = 20$, but $n_s = 21$ is replaced with 3 in Eq. 3.20, because an entry in a $\theta_{i,j}$ matrix in the Θ_n matrix (Eq. 3.19) that turns out to be zero, after multiplication with the disturbance variable $w_{U,n}$ (in $\mathbf{w}_{U,n}$ matrix in Eq. 3.38), can be set to zero to begin with, to further reduce the number of optimisation variables d . We replaced n_s in Eq. 3.20 with 3, because B_U has only 3 non-zero rows in Eq. 3.70, which defines the non-zero rows in the disturbance variable $w_{U,n}$ as $B_U(u_{U,n}^f + w_n)$ (Eq. 3.32). With these values, we get $d = 77$ from Eq. 3.20, and then we increment another variable, on top, corresponding to the h variable, used in the optimisation schemes (Section 3.5).

We used the optimisation schemes in Section 3.5 on the selected river data. In the first scheme we considered $\alpha = 0.1$, i.e. 10% degradation of the performance J is allowed in the second optimisation problem of the scheme. For simulations, we selected the following three situations:

- Situation 1: We considered unregulated inflows from Kiewa and Ovens Rivers were unknown, however, we assumed the demands from irrigation channels and downstream Yarrowonga Weir were exactly known. Forecasts for the flows at Bandiana and Peechelba were based on Eq. 3.39, and we used the forecast models obtained in Appendices 2.C & 2.D. Uncertainties w_B and w_P in the flow forecasts were selected to be independent and identically distributed Gaussian random variables with zero means and 30 ML/Day standard deviations each.
- Situation 2: We considered all unregulated in- and out-flows were unknown except the release from Yarrowonga Weir. Flow forecasts at Bandiana and Peechelba were done the same way as in Situation 1. The demands to irrigation channels were considered to be constants over the finite horizon M . Precisely, we picked two cases: (i) the demand values for the whole horizon were considered to be the first value of the actual demands, (ii) the demand values for the whole horizon were considered to be the average value of the actual demands over the horizon.

- Situation 3: We repeated Situation 1 for the dataset with high inflows from Kiewa and Ovens Rivers. Flow forecasts at Bandiana and Peechelba were done the same way as in Situation 1, however, for the uncertainties w_B and w_P in the flow forecasts, we increased the standard deviations of the random variables from 30 ML/Day to 50 ML/Day for w_B and 200 ML/Day for w_P .

The optimisation problems were solved by running YALMIP [115] over SDPT3 [4].

3.6.2 Performance of Situation 1: (unknown unregulated inflows)

Simulations in this section were carried on the dataset from 7th Oct., 2006 to 20th Dec., 2006. Figs. 3.2 & 3.3 show the performance of Scheme 1. In Fig. 3.2 the black curve shows the actual historical water level, the blue curve shows the response of Scheme 1, and the magenta curve is the simulation of the model in Eq. 3.65 using the historical input data. We can observe that the model is quite accurate since the black and magenta curves are close. The blue curve, which shows the controlled water level obtained from the optimisation scheme, demonstrates that the control scheme not only restricted the water level to be within the limits, but also kept it close to the mean of the limits.

The model simulation seems to drift away from the actual historical water level towards the end. It is because the model has an integrator and the errors accumulate with time. However, it does not affect the MPC scheme much as we initialise the model at each time step based on the measurements at the previous time step.

Fig. 3.3 shows the regulated and the unregulated flows. The top graph shows the regulated flow at Hume (the outcome of Scheme 1), and the corresponding actual flow release. Due to the drop in the release to Mulwala Canal and Yarrawonga Main Channel at 80th and 110th time samples (shown in the lower graph of Fig. 3.3, and separately shown in Fig. 3.4 for clarity), the scheme reduced the release from Hume roughly at the 70th and 100th times samples (see the upper graph of Fig. 3.3). This avoided a rise in the water level which is present in the actual water level with peaks at 90th and 120th times samples (see Fig. 3.2). However, this is not a fair comparison since we had access to the exact future water demands and we adjusted the flow release every 8 hours while the operators only adjusted the flow every 24 hours.

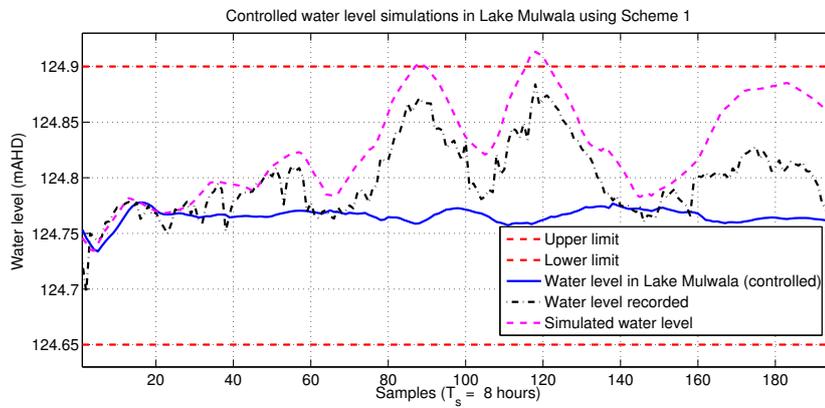


Figure 3.2: Controlled water level using Scheme 1.

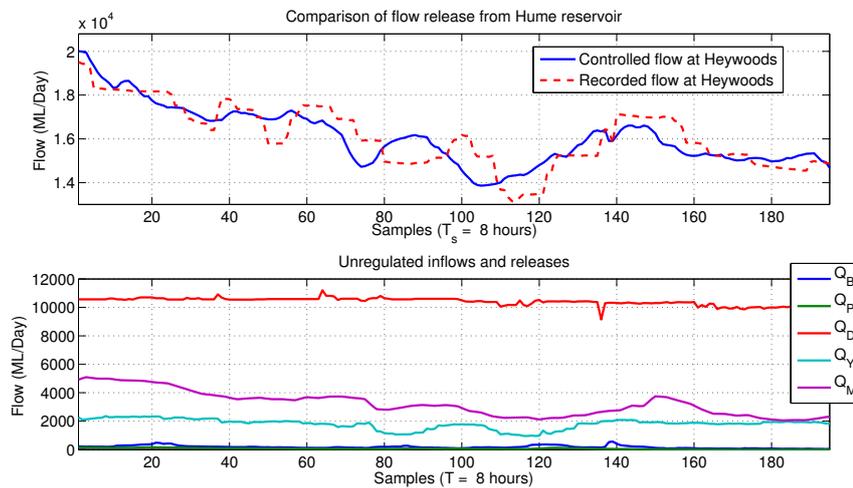


Figure 3.3: Regulated and unregulated flows using Scheme 1.

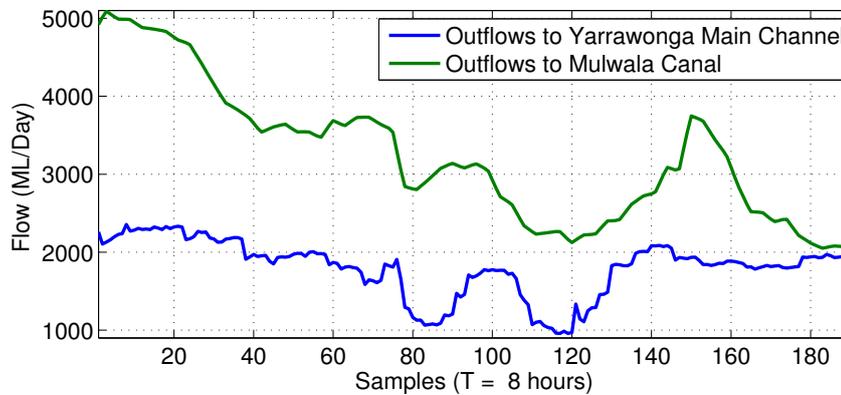


Figure 3.4: Zoomed outflows to Yarrawonga Main Channel and Mulwala Canal.

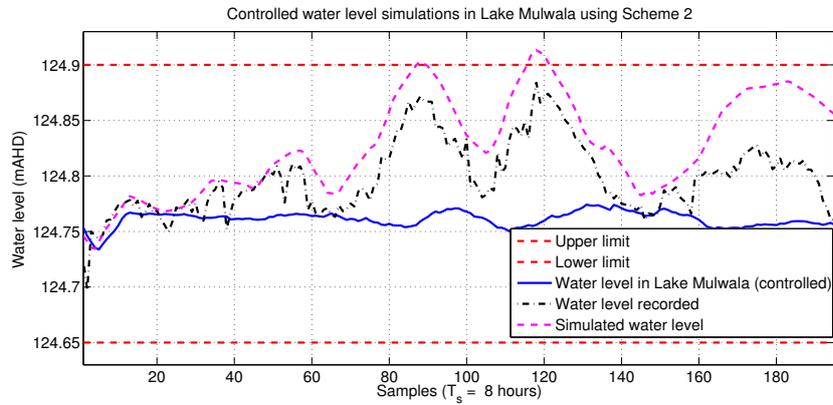


Figure 3.5: Controlled water level using Scheme 2.

Figs. 3.5 & 3.6 show the performance of Scheme 2. This scheme also performed well, with similar results to the previous one. The controlled release from Hume (Fig. 3.6) shows some fluctuations, which are not present in Fig. 3.3. However, we can avoid the fluctuation by considering a non zero \mathbf{R} matrix in the objective criterion $J(\Theta, \Gamma)$ (Eq. 3.52). For this, we selected \mathbf{R} as an $M \times M$ diagonal matrix with $R' = 1 \times 10^{-7}$ in the diagonal. Figs. 3.7 & 3.8 show the performance of Scheme 2 with the above selection of \mathbf{R} matrix in the objective function. In Fig. 3.8, we can see that the fluctuations in the flow release are removed compared to the flow release shown in Fig. 3.6. Moreover, the behaviour of the water level is almost similar in both cases, see Figs. 3.5 & 3.7. In the remaining simulations of this chapter, we selected a zero \mathbf{R} matrix for Scheme 1 and the non-zero \mathbf{R} (specified above) for Scheme 2.

In Scheme 1, the user can manipulate the trade-off (α) between performance and feasibility. On the other hand, Scheme 2 is an automated option and is computationally less expensive if water level crosses the hard limits, since Problem 3.64 is then skipped. Both schemes have performed well in controlling the water level in the lake. In the next subsection we see their performance for the case when demands from the irrigation channels along with the inflows from unregulated rivers were unknown.

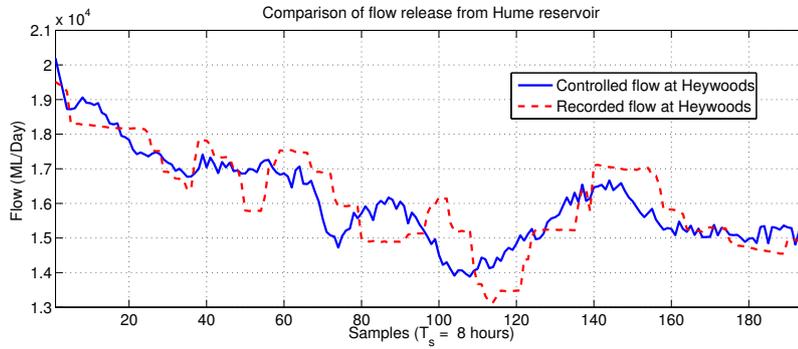
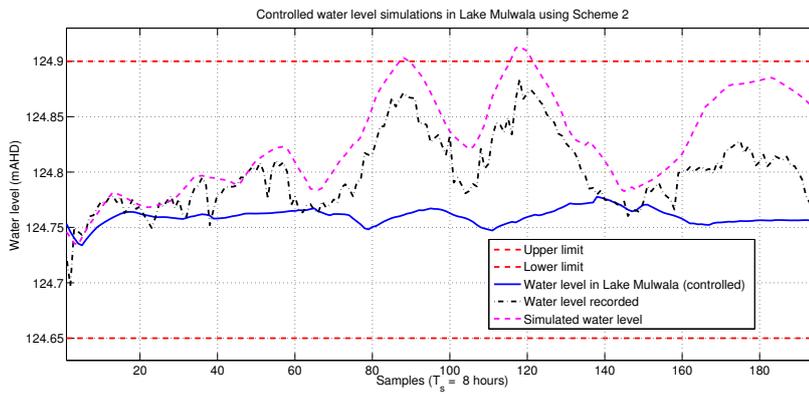
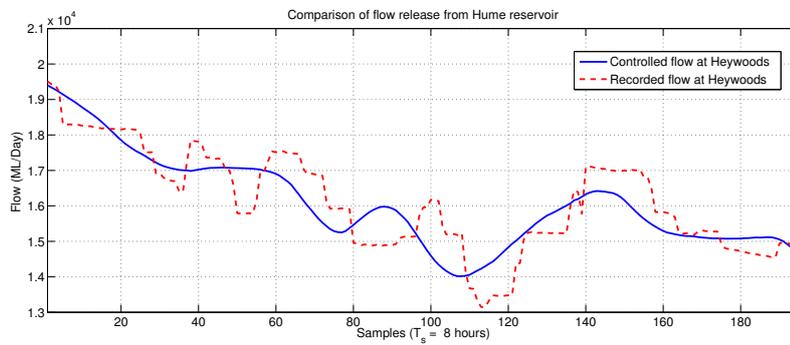


Figure 3.6: Regulated flows using Scheme 2.

Figure 3.7: Controlled water level using Scheme 2 with a non-zero \mathbf{R} matrix in the objective function.Figure 3.8: Regulated flows using Scheme 2 with a non-zero \mathbf{R} matrix in the objective function.

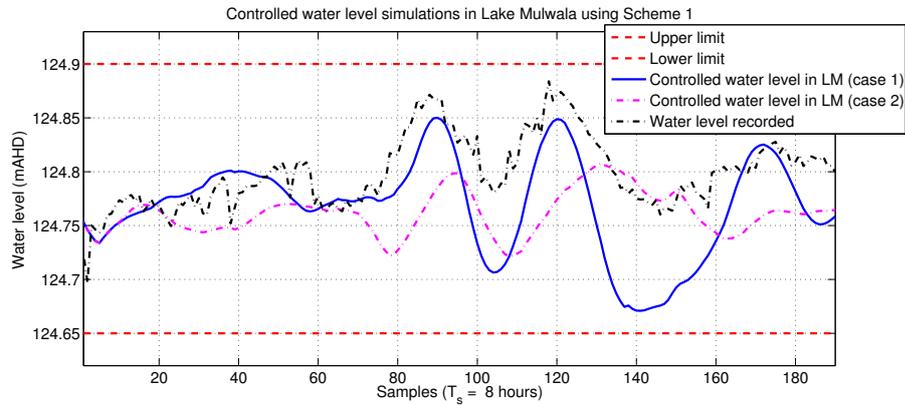


Figure 3.9: Controlled water level using Scheme 1 with unknown demands from irrigation channels.

3.6.3 Performance of Situation 2: (unknown unregulated in- and out-flows)

Simulations in this section were carried on the river dataset from 7th Oct., 2006 to 20th Dec., 2006. As described earlier, in the simulations here, the flow forecasts at Bandiana and Peechelba were obtained from the models in Appendices 2.C & 2.D, and for the unknown demands from Yarrawonga Main Channel and Mulwala Canal, we considered the following two cases,

- Case 1: The forecasted demands for the whole horizon were equal to the first value of the actual demands over the finite horizon ($M = 20$ units). The actual demands are shown in Fig. 3.4. The simulation results corresponding to this case are shown with blue curves in Figs. 3.9, 3.10 & 3.11, 3.12, corresponding to Scheme 1 and Scheme 2 respectively.
- Case 2: The forecasted demands for the whole horizon were equal to the average value of the actual demands over the finite horizon ($M = 20$ units). The simulation results corresponding to this case are shown with magenta dashed-dotted curves in Figs. 3.9, 3.10 & 3.11, 3.12, corresponding to Scheme 1 and Scheme 2 respectively.

The recorded water levels in Lake Mulwala and flows at Heywoods are shown with black dashed-dotted and black dashed curves respectively in the figures.

As expected, the controlled water level in Lake Mulwala (obtained from both schemes, see Figs. 3.9 & 3.11) was not close to the desired water level set-point of 124.775

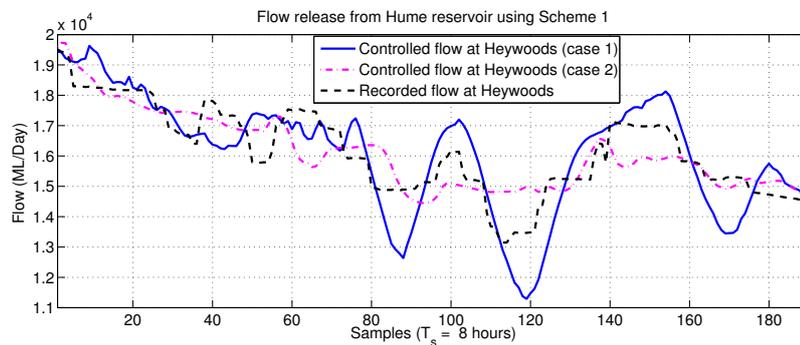


Figure 3.10: Regulated flows using Scheme 1 with unknown demands from irrigation channels.

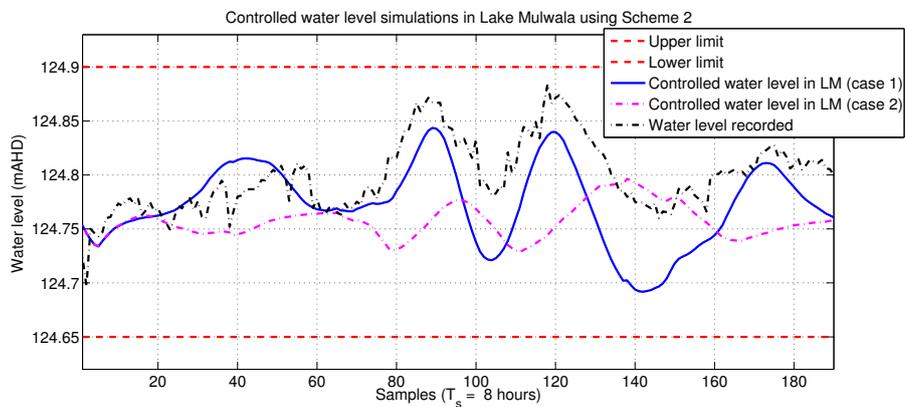


Figure 3.11: Controlled water level using Scheme 2 with unknown demands from irrigation channels.

mAHD. This is because of the errors in the demand forecasts, and especially, the performance of case 1 is poorer compared to case 2, because of larger errors in the forecasts. However, the water levels still stayed within the limits (124.65 and 124.9 mAHD), which shows that the selected control strategy is robust for the selected flow ranges. Figs. 3.10 & 3.12 show the corresponding flow release from Hume reservoir for both cases and control schemes. The flows adhered to the constraints well, and again as expected, the performance of case 2 seems to be better than case 1, as shown in the figures.

The above cases were purposely selected with poor forecasts to see the performance of the developed schemes in unfavourable situations. In practice, river operators are usually aware of the demand profiles for the next 3 to 4 days. To demonstrate the performance of the control schemes with a more realistic demand forecast from the irrigation channels,

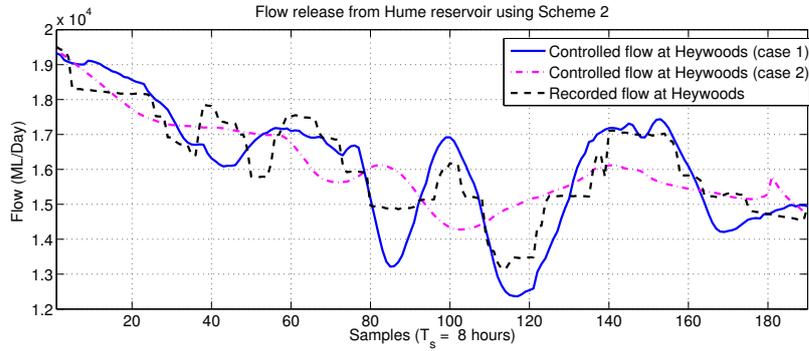


Figure 3.12: Regulated flows using Scheme 2 with unknown demands from irrigation channels.

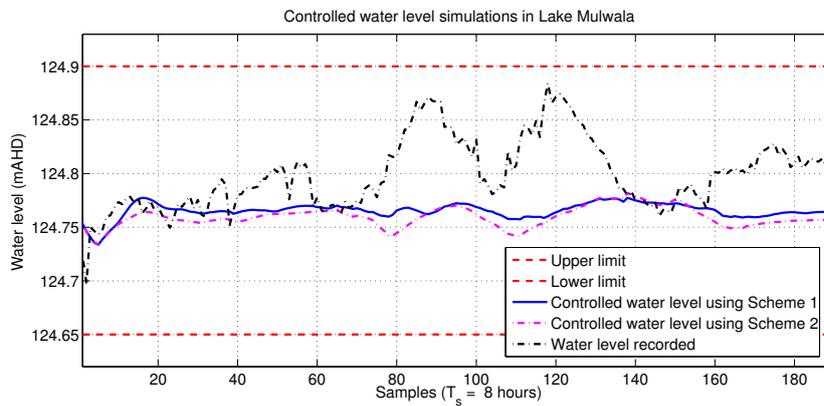


Figure 3.13: Controlled water level in Lake Mulwala with a realistic forecast of demands from irrigation channels.

we considered the following case. We assumed that we knew the demand profile for the first 10 steps (about $3\frac{1}{2}$ days) of the finite horizon, and for the remaining 10 steps, we used the average value of the actual demands on those remaining 10 steps, for each step. Figs. 3.13 & 3.14 show the performance of both control schemes, and we can see that the schemes performed almost similar to each other. Fig. 3.13 show the controlled water level in Lake Mulwala, which ideally stays close to 124.775 mAHD, throughout the simulations, and the corresponding regulated flow release at Heywoods is shown in Fig. 3.14. From the simulation results, we can say that the performance of the control schemes, with a realistic demand forecast, is acceptable for the given river.

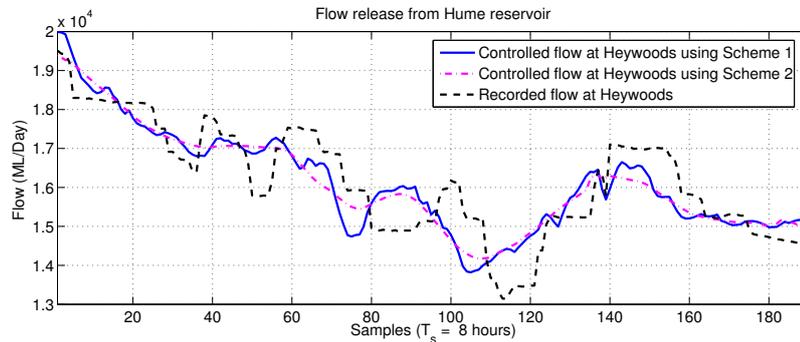


Figure 3.14: Regulated flows at Heywoods with a realistic forecast of demands from irrigation channels.

3.6.4 Performance of Situation 3: (high unregulated inflows)

In this section we examine the control of water level in Lake Mulwala when the unregulated inflows (from Kiewa and Ovens Rivers) are above their normal operating range. The schemes are designed to provide feasible solutions for a wide range of inflows, but we are interested to see whether their control performance (found from the value of J) is also acceptable or not.

We picked the dataset from 19th Sep., 2001 to 16th Nov., 2001, sampled at $T_s = 8$ hours. Fig. 3.15 shows the inflows and the outflows, and we can see that the unregulated inflows from Kiewa and Ovens Rivers (i.e. at Bandiana (B) and Peechelba (P)) are high compared to Fig. 3.4. Again, we use the forecast models (of flows at Bandiana and Peechelba) in Eqs. 2.47 & 2.52. Fig. 3.16 shows the simulations of the flows at Bandiana and Peechelba for the selected dataset. The blue curves show the actual recorded data and the red dashed curves show the simulation of the forecast models. The simulation results of the flows at Bandiana are good, however, for the flows at Peechelba, the model picks the trends well, but at flow peaks, the simulated flows are off by several hundred ML/Day from the actual flows. Due to high inflows, we expect high uncertainty in their forecasts, and because of this reason we increased the standard deviations of the additive Gaussian distributed noise terms w_B and w_P to 50 ML/Day and 200 ML/Day respectively.

Figs. 3.17 & 3.18 show the performance of Scheme 1. Fig. 3.17 shows water level in Lake Mulwala. The blue curve in the figure shows the controlled water level, the black curve shows the actual recorded water level and the magenta curve shows the simulation

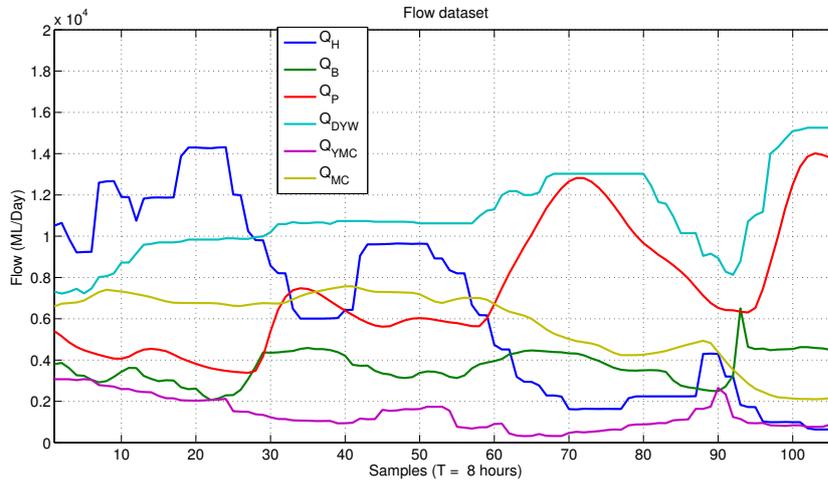


Figure 3.15: Inflows and outflows in the dataset.

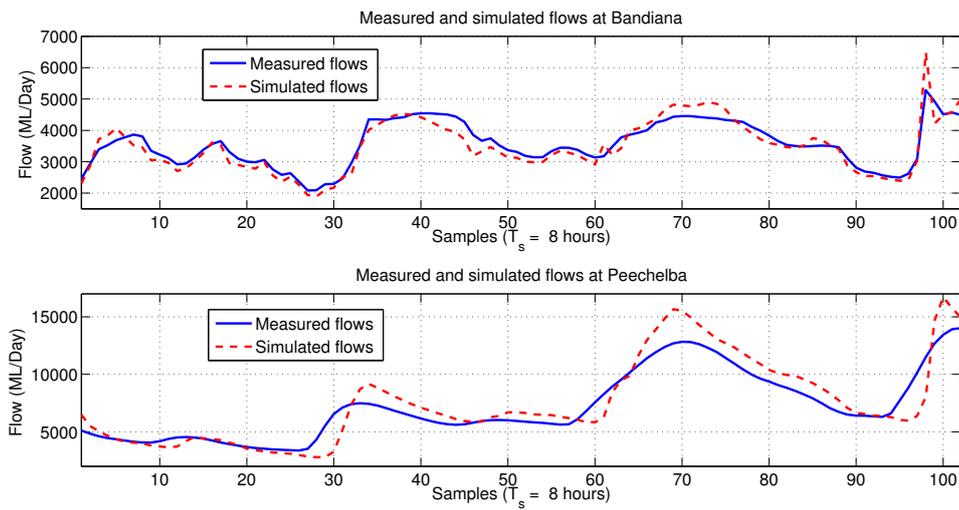


Figure 3.16: Simulated flows at Bandiana and Peechelba.

of the model in Eq. 3.65 using the measured input data. The model performed reasonably well and picked the main trends in the data. Again, the model simulation seems to drift away from the actual recorded data, but this is because of the integrator in the model that accumulates errors. However, it does not affect the MPC schemes as we initialise the model at each time step, based on the measurement at the previous time step.

The performance of the controlled water level in Fig. 3.17 is better than the actual recorded data, but it is not a desirable one, because the water level crossed the upper allowed limit (124.9 mAHD) twice. It crossed only a little at 45th time sample, however, it stayed outside the allowed limit from 75th to 97th time sample. This can be explained from the corresponding flow release at Heywoods, shown in Fig. 3.18. During this time window, the flow release at Heywoods stayed at the lower flow limit (i.e. 2,500 ML/Day). However, the recorded flow data (red curve) show that the operators reduced the flows even below 2,500 ML/Day and they hit a level of 600 ML/Day, which are outside the normal operating range and they correspond to flood operating range, which is a regime outside the scope of this chapter. Even with such actions, the recorded water level in Fig. 3.17 stayed higher than what we obtained from the control scheme. It can be explained by looking at the events in Figs. 3.17 & 3.18,

- The peak of the controlled flow release at around 5th time sample, which is above the recorded flow release, caused the controlled water level to rise above the recorded water level between 12th and 30th time samples, but the controlled water level dropped below the recorded water level after that. This is achieved by decreasing the controlled flows at the maximum allowed rate, i.e. -500 ML/Day, from roughly 6th to 16th sampling instant.
- The recorded flow release stayed 2,000 ML/Day to 4,000 ML/Day higher than the controlled flow release between 15th and 25th sampling time, which caused the recorded water level to cross the upper limit (124.9 mAHD) and it got roughly 8 cm higher than the controlled water level.
- Furthermore, higher flow release in the recorded data between 40th and 50th time samples kept the recorded water level above the controlled water level and the

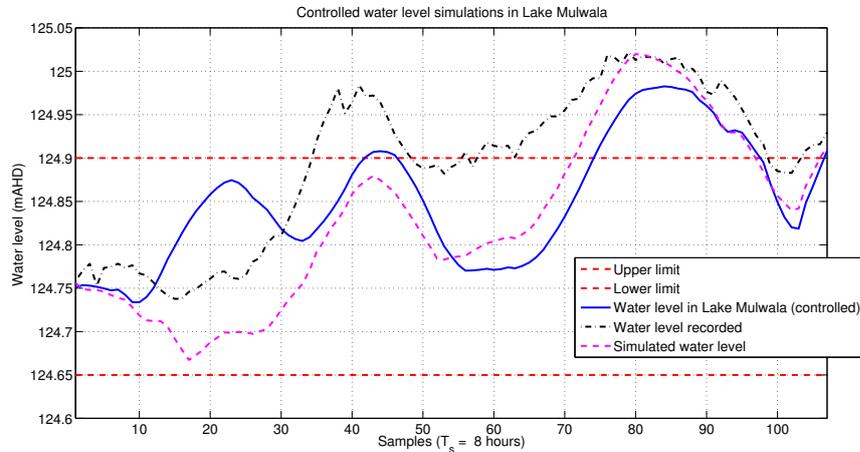


Figure 3.17: Controlled water level using Scheme 1.

upper limit (124.9 mAHD).

- Finally, the high unregulated inflow at Peechelba around the 70th time sample (shown in Fig. 3.15) caused the controlled and the recorded flow release to be reduced at the maximum allowed rate (-500 ML/Day), but this was not enough to prevent the water level to cross the upper limit. The rate of increase of the controlled water level is higher as the constraint on the lower limit of the corresponding flow release at Heywoods was adhered to in the optimisation problem. However, the operators decreased the flow below the lower limit at that time to avoid rapid increase in the recorded water level, but still it increased due to the high inflow at Peechelba.

Figs. 3.19 & 3.20 show the performance of Scheme 2 (with $\mathbf{R} = \mathbf{0}$) on the same dataset with the same control parameters as used in Scheme 1. The performance of the scheme is similar to Figs. 3.17 & 3.18, and can be explained the same way as above. The only difference is that in this case, the controlled water level exceeded the upper limit a little less between 75th and 95th time samples as shown in Fig. 3.19, as compared to Fig. 3.17. The reason is that the water level exceeded the limit a little more between 40th and 50th time samples, the controlled flows at Heywoods were dropped further at around 53rd time samples, compared to what they were in Fig. 3.18.

The increase in the flow release at Heywoods from 77th to 86th time samples in Fig.

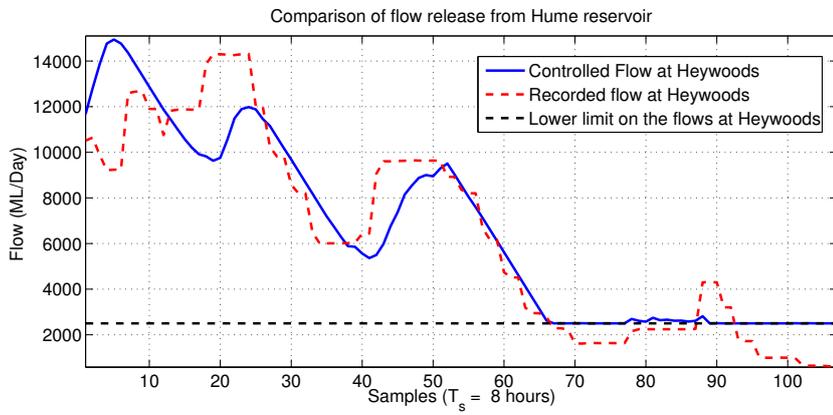


Figure 3.18: Regulated flows at Heywoods using Scheme 1.

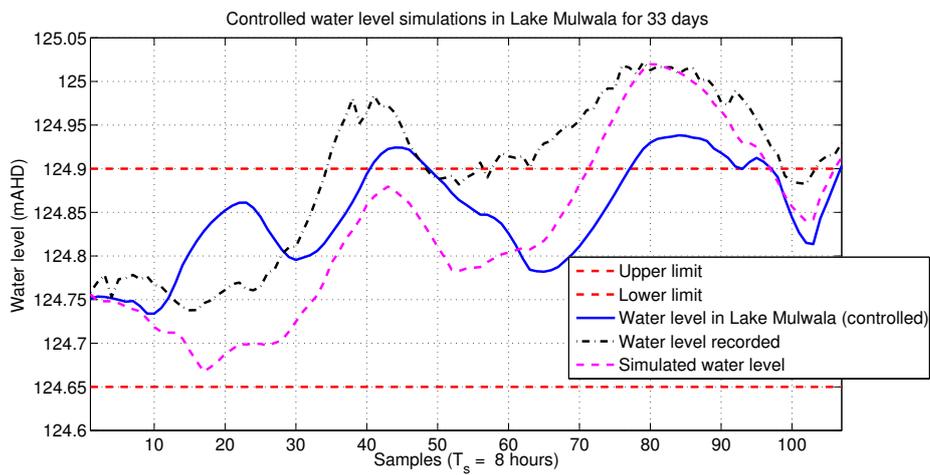


Figure 3.19: Controlled water level using Scheme 2.

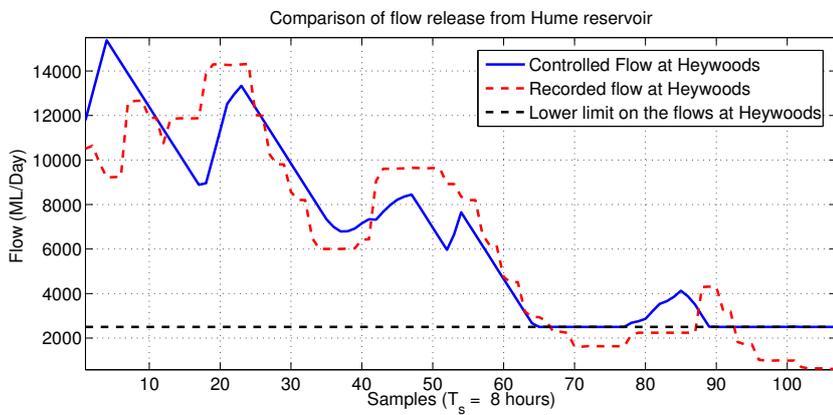


Figure 3.20: Regulated flows at Heywoods using Scheme 2.

3.20 looks suspicious, if compared with the water levels in Lake Mulwala during this period in Fig. 3.19, which are already above the upper limit (124.9 mAHD). This behaviour is also evident in the recorded flows in Fig. 3.20 (shown with the red dashed curve), from 87th to 90th time samples. However, this can be explained by taking the time delays in the system into account and observing the flows and water levels in Figs. 3.15, 3.16, 3.19 & 3.20 during this time period. The optimisation problem at 77th time step anticipated the rise in the rapid release from Yarrawonga Weir from 92nd time step onward (see Fig. 3.15). Activities at Yarrawonga Weir are 9 steps ahead in time due to the time delays, and the forecasts in the finite horizon, which is additional 20 steps, helped the controller to anticipate the rapid release from the weir. During this period, the inflows from Ovens Rivers were also forecasted to decrease till roughly 96th time sample (as in Fig. 3.16, the red dashed curve). Therefore to compensate the expected sudden fall in water level in the lake, the flow release from Hume was increased, and even with it, we can still see the rapid decrease in water level from 95th to 103rd time samples. However, the flows at Heywoods were later decreased again to the lowest allowed value (2,000 ML/Day), because the water level in the lake got increased due to the rapid increase in the forecasted inflow from Ovens River from 97th time sample onwards, as shown in Fig. 3.16.

To sum up, the control schemes performed better than the recorded data but there is a need of a control strategy which can handle unregulated flows in a better way, which should be able to mitigate flood risks and provide probabilistic guarantees on the solution. Also, we found that when water level is expected to exceed the upper limit (e.g. 124.9 mAHD in our case), we can reduce the flow release from the storages below the lower limit to avoid major damage. With this knowledge, we will propose and use a flood risk mitigation strategy in the next two chapters.

This page intentionally left blank.

Appendix

3.A Additional topics on control policies

3.A.1 A control policy which accommodates forecasts

The forecast models in Section 3.3.2 can improve a control policy. As mentioned in Section 3.2.2, the parameters in the Θ_n matrix (Eq. 3.16) corresponding to the disturbances which are not measured or computable, at a given time n , should be set to zero. However, with the availability of the forecast models, the $w_{U,n}$ entries in the $\mathbf{w}_{U,n}$ matrix in Eq. 3.16 can be replaced by $B_U u_{U,n}^f$, where the flow forecasts $u_{U,n}^f$ are given by Eqs. 3.39 & 3.40. Also, we can use the forecasts of all future disturbances in the finite horizon M to define a control policy. Based on Eq. 3.16 we have

$$\mathbf{u}_n = \Theta_n \mathbf{w}_{U,n} + \Gamma_n + \Theta'_n \begin{bmatrix} B_U u_{U,n}^f & B_U u_{U,n+1}^f & \cdots & B_U u_{U,n+M-1}^f \end{bmatrix}^\top, \quad (3.66)$$

where Θ_n and Γ_n matrices are as before (Eq. 3.16), and

$$\Theta'_n = \begin{bmatrix} \theta'_{n,0} & \theta'_{n,1} & \cdots & \theta'_{n,M-1} \\ \mathbf{0} & \theta'_{n+1,1} & \cdots & \theta'_{n+1,M-1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \theta'_{n+M-1,M-1} \end{bmatrix},$$

provided all $\theta_{i,j}$ entries in the lower triangle of Θ_n matrix (Eq. 3.16) are non-zero, otherwise, the corresponding $\theta'_{i,j}$ entries in the lower triangle of matrix Θ'_n are kept non-zero, and are considered as optimisation parameters. A control policy based on Eq. 3.66 in-

creases the number of parameters more than double the parameters in Eq. 3.16, and it is computationally more expensive to use such a control policy in an optimisation problem. In this chapter we consider the control policy in Eq. 3.16, however, the policy in Eq. 3.66 is also applicable. Moreover, the policy does not affect the convexity of the control optimisation problems in Sections 3.4 & 3.5 with respect to the optimisation variables $(\Gamma_n, \Theta_n, \Theta'_n)$

3.A.2 Incorporating an integral action

A typical control objective for a river is to maintain a water level or flow between an upper and lower limit in a storage or at a particular point along the river. However, it can be required to keep a water level or flow at a specific set-point. For such cases, we can add integral action to the control policy to avoid steady state errors. To achieve this, we augment an extra state with the state space system (Eq. 3.28). The purpose of the state is to accumulate (integrate) the deviations in the water level from a reference point, say y_{ref} , and then the state value can be minimised in an optimisation problem. From Eq. 3.28, we have

$$x_{n+1} = Ax_n + Bu_n. \quad (3.67)$$

We introduce a state that accumulates the deviation of water level from the reference point

$$\xi_{n+1} = \xi_n + Cx_n - y_{\text{ref}}, \quad (3.68)$$

where the dimension of the matrix C is $1 \times n_s$ (for the single water level output), and $C = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \end{bmatrix}$, since the first state in Eq. 3.67 is the water level. This gives the augmented system

$$x_{\text{au},n+1} = \begin{bmatrix} x_{n+1} \\ \xi_{n+1} \end{bmatrix} = \begin{bmatrix} A & \mathbf{0} \\ C & 1 \end{bmatrix} \begin{bmatrix} x_n \\ \xi_n \end{bmatrix} + \begin{bmatrix} B & \mathbf{0} \\ \mathbf{0} & -1 \end{bmatrix} \begin{bmatrix} u_n \\ y_{\text{ref}} \end{bmatrix},$$

where the subscript ‘au’ represents augmented states. The above equation can be written as

$$x_{\text{au},n+1} = A_{\text{au}}x_{\text{au},n} + B_{\text{au}} \begin{bmatrix} u_n & y_{\text{ref}} \end{bmatrix}^T. \quad (3.69)$$

We can treat Eq. 3.69 the same way as we treated Eqs. 3.1 & 3.28 in this chapter. Also, the augmentation does not affect convexity, and is applicable to a general river.

The above formulation can suffer with integrator windup problems when constraints become active (i.e. when actuators hit a saturation). However, as described earlier, we do not require set-point tracking in rivers in general, and therefore an integral action is not needed. Nevertheless, if its necessary, it is advisable to add an anti-windup mechanism to the integral action formulation, see [121].

3.B Proof of Corollary 3.1

Fig. 3.21 (a) shows an illustration of an N dimensional set ‘ \mathcal{W}^N ’. Each element of the set is a possible ‘multiple sample extraction’: $(\mathbf{w}_{U,n}^1, \mathbf{w}_{U,n}^2, \dots, \mathbf{w}_{U,n}^N)$, where $\mathbf{w}_{U,n}^k$ is an Mn_s dimensional vector, where n_s is the number of states. Assume the blue portion of the set (shown with blue circles) consists of multi-samples that give a solution with $h_1 > \bar{h}$, and the black portion (shown with black squares) consists of those that give a solution with $h_1 \leq \bar{h}$. The portions are separated with a blue dashed line. The dashed region in Fig. 3.21 (a) shows a bad set of multi-samples that give a solution to the scenario version of Problem (3.59) which is not feasible for Problem (3.59). According to the scenario theory [117], the probability measure of this region is $\leq \beta$. Note that the dashed region can be in both the blue and black portions of \mathcal{W}^N . On the other hand, our randomised solution can fall in this bad set only if $h_1 > \bar{h}$, because if $h_1 \leq \bar{h}$, then we proceed and solve the scenario version of Problem (3.60). The green dashed region in Fig. 3.21 (b) shows a new bad set corresponding to the multi-samples that give a solution that satisfies $h_1 \leq \bar{h}$, but is not feasible for Problem (3.60). According to [102] & [117], this bad set has also a probability β at most. In any case, when we solve the scenario version of the cascade of Problems (3.59) & (3.60), the bad set, where either $h_1 > \bar{h}$ and the solution is not feasible for Problem (3.59) or $h_1 \leq \bar{h}$ and the solution is not feasible for Problem (3.60), is at most

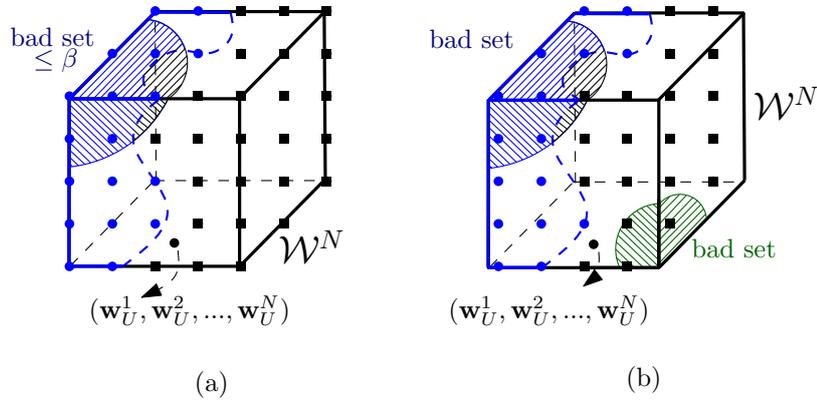


Figure 3.21: Interpretation of the scenario theorem applied to scheme 3.

the union of the blue (say 'A') and green (say 'B') dashed regions; then, we have

$$\mathbb{P}(A \cup B) \leq \mathbb{P}(A) + \mathbb{P}(B) \leq 2\beta.$$

i.e. the solution is feasible with confidence at least $1 - 2\beta$.

□

3.C State space model of the water level in Lake Mulwala

A discrete time state space model of the water level in Lake Mulwala (y_{LM}), obtained from the procedure described in Section 3.3.1, using the model in Eq. 3.65, is given below.

The model has $n_s = 21$ states with the step size of 8 hours.

$$x_{n+1} = \begin{bmatrix} 1 & c_H & \mathbf{0}_{1 \times 8} & c_B & \mathbf{0}_{1 \times 8} & c_P & 0 \\ \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 1} & \mathbf{I}_{8 \times 8} & \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 8} & \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 1} \\ 0 & 0 & \mathbf{0}_{1 \times 8} & 0 & \mathbf{0}_{1 \times 8} & 0 & 0 \\ \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 8} & \mathbf{0}_{8 \times 1} & \mathbf{I}_{8 \times 8} & \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 1} \\ 0 & 0 & \mathbf{0}_{1 \times 8} & 0 & \mathbf{0}_{1 \times 8} & 0 & 0 \\ 0 & 0 & \mathbf{0}_{1 \times 8} & 0 & \mathbf{0}_{1 \times 8} & 0 & 1 \\ 0 & 0 & \mathbf{0}_{1 \times 8} & 0 & \mathbf{0}_{1 \times 8} & 0 & 0 \end{bmatrix} x_n + \begin{bmatrix} 0 & 0 & 0 & c_{DYW} & c_{YMC} & c_{MC} \\ \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 1} \\ 1 & 0 & 0 & 0 & 0 & 0 \\ \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 1} \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} Q_H \\ Q_B \\ Q_P \\ Q_{DYW} \\ Q_{YMC} \\ Q_{MC} \end{bmatrix},$$

$$x_{n+1} = Ax_n + Bu_n,$$

where the B matrix can be partitioned into regulated and unregulated parts, corresponding to regulated and unregulated flows as,

$$x_{n+1} = Ax_n + B_C u_{C,n} + B_U u_{U,n},$$

where B_C corresponds to flows at Heywoods and B_U corresponds to the rest unregulated flows, i.e.

$$B_C = \begin{bmatrix} 0 \\ \mathbf{0}_{8 \times 1} \\ 1 \\ \mathbf{0}_{8 \times 1} \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad B_U = \begin{bmatrix} 0 & 0 & c_{DYW} & c_{YMC} & c_{MC} \\ \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 1} \\ 0 & 0 & 0 & 0 & 0 \\ \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 1} & \mathbf{0}_{8 \times 1} \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (3.70)$$

The output (water level in Lake Mulwala) equation is given below,

$$y_{LM,n} = \begin{bmatrix} 1 & \mathbf{0}_{1 \times 20} \end{bmatrix} x_n.$$

The elements in the matrices in the above equations are either scalars or matrices (dimensions are provided in the subscripts), where $c_H = 6.20 \times 10^{-7}$, $c_B = 9.63 \times 10^{-7}$, $c_P = 5.89 \times 10^{-7}$, $c_{DYW} = -6.13 \times 10^{-7}$, $c_{YMC} = -9.73 \times 10^{-7}$ and $c_{MC} = -6.67 \times 10^{-7}$.

Chapter 4

A Randomised Approach to Multiple Chance-Constrained Problems

To effectively control a system, the system's physical and operational constraints are very important, and they must be taken into account, e.g. in a control optimisation problem. However, in some systems, a hard constraint is not advisable, as it might not only degrade the performance of the solution, but it can also cause infeasibility of the optimisation problem.

We dealt with a similar problem in Chapter 3, where we developed a Stochastic MPC based control strategy for rivers with probabilistic constraints on water level and feedback based control policies to avoid infeasibilities. We selected such a strategy because the system disturbances, which are composed of unregulated in- and out-flows, cannot be bounded in a sensible way, and they mostly act additively on the water level. Depending on a system's requirements, these probabilistic constraints (or chance-constraints) are required to be satisfied with a certain probability, and it is possible that a system can have chance-constraints with different probability of assurance requirements. An optimisation problem with such multiple chance-constraints is known as Multiple Chance-Constrained optimisation Problem (M-CCP) [122], and in this chapter we propose an algorithm to find approximate solutions to such problems.

As an example, the control Problem (3.55) deals with normal river operations only, which include keeping flows, water levels and their change within acceptable limits. The control problem does not take flood avoidance into account, and the flood risk mitigation should also be done with a higher assurance probability. An additional chance-constraint in Problem (3.55), related to flood risks mitigation can serve this purpose well.

In general, a system can have two modes of control operations: normal operations and operations for exceptional situations (e.g. flood avoidance in rivers). The normal operations are usually carried out always, however, at exceptional situations, we often give up on them to avoid damage (e.g. minimum flow release from storages when there is a high flood risk). Such a control problem can be represented as an M-CCP with two chance-constraints: one related to the normal operations and the other related to risk mitigation. For risk mitigation, we also need a way to measure the risk. There are several risk measuring tools available in the literature, we briefly discuss Value-at-Risk (VaR) and Conditional Value-at-Risk (CVaR) [86] in this chapter, and we use VaR in the control problem, because it can conveniently describe the second chance-constraint.

As described in Chapter 3, the scenario-based approach (Section 3.2.4, [27, 117]) can be used to find approximate solutions to chance-constrained problems, and the approach can also be extended to M-CCPs [122]. However, due to the high probability assurance requirement of the risk mitigation chance-constraint, it is expected that the scenario approach will add a very large number of constraints in the optimisation problem, and the problem can become computationally very expensive.

In this chapter we propose an *Optimisation and Testing algorithm* that uses the scenario approach together with ideas borrowed from validation set methods [118, 123–125] to find approximate solutions to M-CCPs. The algorithm works in three steps: *optimisation*, *testing* and *improving*. In the *optimisation* step, the algorithm solves an optimisation problem with the chance-constraint associated with the normal operations only. In the *testing* phase, the algorithm tests the solution against the chance-constraint associated with risk mitigation, which is computationally very inexpensive. No further optimisation is performed if the test is passed, otherwise, the solution is improved in a computationally cheap way by solving a one-dimensional scenario problem to satisfy the latter chance-constraint in the *improving* phase. The proposed algorithm is non-iterative, it is a computationally effective way to solve M-CCPs, and it is extendible to problems with more than two chance-constraints. A basic idea of such an extension is also presented towards the end of this chapter.

The chapter is organised as follows. In Section 4.1 we discuss preliminary concepts

which are used in this chapter, and then we state the M-CCP with two chance-constraints. A discussion on some existing ideas to solve M-CCPs are given in Section 4.2. The *Optimisation and Testing algorithm* to solve such M-CCPs is presented in Section 4.3. The chapter concludes with an algorithm to extend the *optimisation* and *testing* idea to solve M-CCPs with any finite number of chance-constraints in Section 4.5.

4.1 Preliminary concepts and problem statement

In this section we describe some preliminary concepts and give definitions, which are later used to define some commonly used risk measures and the Multiple Chance-Constrained optimisation Problem (M-CCP), that we seek to solve in this chapter.

4.1.1 Loss function and loss distribution

We consider a system \mathcal{S} , which is described by the following state space model,

$$x_{n+1} = Ax_n + Bu_n + w_n, \quad (4.1)$$

$$y_n = Cx_n, \quad (4.2)$$

where $x \in \mathbb{R}^{n_s}$ is a state vector, $u \in U \subset \mathbb{R}^d$ is a vector of input variables, $w \in W \subset \mathbb{R}^{n_s}$ is a vector of uncertain variables, and $y \in \mathbb{R}^p$ is a vector of outputs of the system \mathcal{S} . We assume that the system \mathcal{S} has normal control operations and some control operations related to risk mitigation. Furthermore, we assume the disturbances w_n, w_{n+1}, \dots are independent and identically distributed.

A *loss function* $g(u, w)$, related to the risks associated with \mathcal{S} , is a function of the input variables u and the uncertain variables w . The loss function $g(u, w)$ may or may not be the same as the (loss) objective function related to the normal operations of the system, say $J(u, w)$. For generality, we consider the loss function $g(u, w)$ to be different from $J(u, w)$ in this work. Furthermore, we intend to keep the loss function $g(u, w)$ in the constraints of the control optimisation problem, because we want to be concerned about it only when a risk related to \mathcal{S} appears. We do not want to degrade the control performance by in-

incorporating $g(u, w)$ alongside the objective function of the optimisation problem ($J(u, w)$), and being overly cautious about risks all the time.

We assume the uncertainty w_n is governed by a probability measure \mathbb{P}_W on the set W , and the distribution $\psi(u, \zeta)$, corresponding to the loss function $g(u, w)$, can be described as

$$\psi(u, \zeta) = \mathbb{P}_W \{w \in W | g(u, w) \leq \zeta\}, \quad \forall u \in U, \quad (4.3)$$

where ζ quantifies loss. $\psi(u, \zeta)$ is described as the *loss distribution*, as it gives the probability that the loss $g(u, w)$ is less than or equal to ζ .

4.1.2 Risk measures

There are various risk measures available in the literature. In this work we discuss Value-at-Risk (VaR) and Conditional Value-at-Risk (CVaR), as they can be easily incorporated in an optimisation problem. Especially, VaR suits our intended M-CCP framework, because it can be incorporated as a chance-constraint. VaR and CVaR are the commonly used risk measures in banking and finance, e.g. see [86, 126] and the references there-in.

Value-at-Risk (VaR)

α -VaR ($\zeta_\alpha(u)$) is the α^{th} percentile of a loss distribution, where $\alpha \in (0, 1)$. It is the smallest value, say ζ^* , such that the loss exceeds ζ^* with a probability at most $1 - \alpha$. α -VaR can be expressed as [86],

$$\zeta_\alpha(u) = \min\{\zeta | \psi(u, \zeta) \geq \alpha\}, \quad (4.4)$$

corresponding to the decision u , where $\psi(u, \zeta)$ is a loss distribution defined in Eq. 4.3. Fig. 4.1 shows α -VaRs corresponding to the two distribution curves shown in red and blue. α -VaR corresponding to the red curve is smaller than the one corresponding to the blue curve. The distribution curves shown in Fig. 4.1 are both continuous and strictly increasing. In such cases VaR can be conveniently and uniquely determined by solving the following expression for ζ

$$\psi(u, \zeta) = \alpha. \quad (4.5)$$

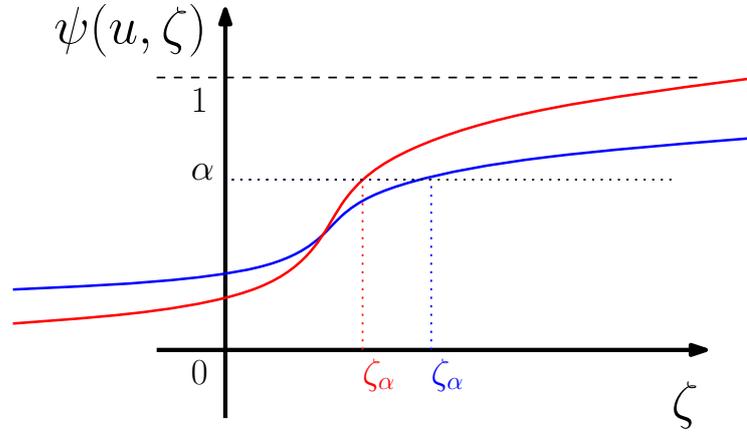


Figure 4.1: An example of loss distribution function.

However, when a probability distribution is non-continuous, e.g. an empirical distribution derived from operational data, then it can have jumps and constant segments. In such cases, we do not expect Eq. 4.5 to have a (unique) solution. Moreover, the convexity of Eq. 4.4, with respect to u , is not guaranteed in general, and it can have many extremums, especially, for discrete loss-distributions [86]. However, VaR can be incorporated in a problem as a chance-constraint, e.g. consider the following constraint on α -VaR,

$$\zeta_\alpha(u) \leq \gamma, \quad (4.6)$$

where γ is a user chosen loss value corresponding to the user chosen $\alpha \in (0, 1)$, γ should be selected carefully, and it should not be too conservative to affect the system during normal operations. We can reformulate the above VaR constraint as the following chance-constraint

$$\mathbb{P}_w \{w \in W : g(u, w) \leq \gamma\} \geq \alpha. \quad (4.7)$$

Eq. 4.7 is a chance-constraint, and it can be added to a chance-constrained optimisation problem to form an M-CCP, for which we seek approximate solutions in this chapter.

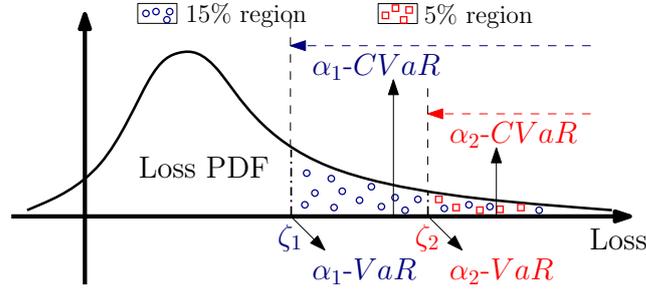


Figure 4.2: An example of a probability density function of loss for a given decision.

Conditional Value-at-Risk (CVaR)

α -CVaR ($\phi_\alpha(u)$) is the expected value of the α -tail distribution of a loss function $g(u, w)$ [86]. The α -tail distribution is given by

$$\psi_\alpha(u, \zeta) = \begin{cases} 0 & \forall \zeta < \zeta_\alpha(u), \\ \frac{\psi(u, \zeta) - \alpha}{1 - \alpha} & \forall \zeta \geq \zeta_\alpha(u), \end{cases} \quad (4.8)$$

for a loss distribution $\psi(u, \zeta)$ (Eq. 4.3). It is also described as the mean loss beyond α -VaR. Fig. 4.2 shows an example of a loss Probability Density Function (PDF), with α -VaRs and α -CVaRs corresponding to two regions with high losses. The portions in the figure with blue circles and red squares are the regions with high losses, and the losses occur with probability 0.15 and 0.05 respectively. ζ_1 and ζ_2 are the VaRs with $\alpha_1 = 0.85$ and $\alpha_2 = 0.95$ corresponding to the portions with blue circles and red squares, and α_1 -CVaR and α_2 -CVaR are the mean losses in those regions.

In addition to CVaR, the functions “upper CVaR” (CVaR⁺) and “lower CVaR” (CVaR⁻) [86] can be defined as,

$$\begin{aligned} \alpha\text{-CVaR}^+ &= \phi_\alpha^+(u) = \mathbb{E}\{g(u, w) | g(u, w) > \zeta_\alpha(u)\}, \\ \alpha\text{-CVaR}^- &= \phi_\alpha^-(u) = \mathbb{E}\{g(u, w) | g(u, w) \geq \zeta_\alpha(u)\}. \end{aligned}$$

In parallel to the definition of α -CVaR with reference to Eq. 4.8, we can define α -CVaR⁺

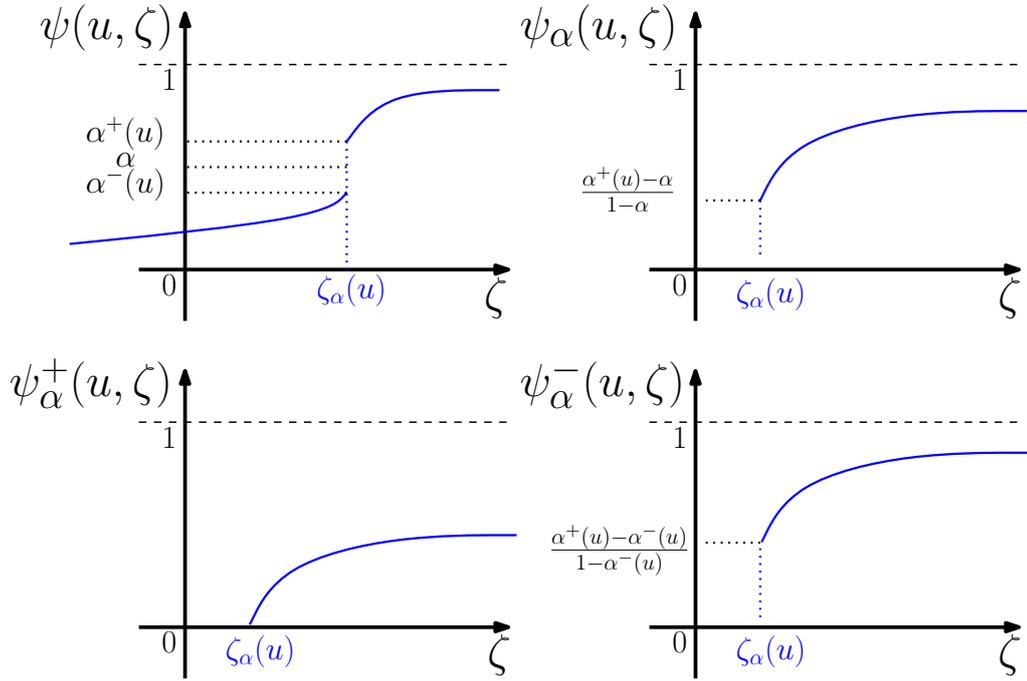


Figure 4.3: A non-continuous loss distribution function and its normalised tails.

and α -CVaR⁻ as the mean of the following two tail distributions respectively

$$\psi_\alpha^+(u, \zeta) = \begin{cases} 0 & \forall \zeta < \zeta_\alpha(u), \\ \frac{\psi(u, \zeta) - \alpha^+(u)}{1 - \alpha^+(u)} & \forall \zeta \geq \zeta_\alpha(u), \end{cases}$$

$$\psi_\alpha^-(u, \zeta) = \begin{cases} 0 & \forall \zeta < \zeta_\alpha(u), \\ \frac{\psi(u, \zeta) - \alpha^-(u)}{1 - \alpha^-(u)} & \forall \zeta \geq \zeta_\alpha(u), \end{cases}$$

where $\alpha^+(u) = \psi(u, \zeta_\alpha(u)) = \mathbb{P}_w\{w \in W | g(u, w) \leq \zeta\}$ and $\alpha^-(u) = \psi(u, \zeta_\alpha(u)^-) = \mathbb{P}_w\{w \in W | g(u, w) < \zeta\}$ define the upper and lower ends of a discrete step in the loss distribution, for more details, see [86]. Fig. 4.3 shows a non-continuous loss distribution $\psi(u, \zeta)$ and its tails corresponding to CVaR, CVaR⁺ and CVaR⁻. Moreover, $\text{VaR} \leq \text{CVaR}^- \leq \text{CVaR} \leq \text{CVaR}^+$ [86].

Discussion

CVaR is convex with respect to u , provided the loss function $g(u, w)$ is convex with respect to u [86]. Moreover, CVaR accounts for losses that correspond to VaR and the losses exceeding VaR. On the other hand, VaR does not take into account the losses exceeding it.

However, system operators are concerned about events with high risks, and in many applications, the loss functions are selected in such a way that the losses associated with such events usually correspond to high α^{th} percentiles of the loss distributions, e.g. $\alpha = 0.999$. In such cases, having a constraint on the high α^{th} percentile, or having a constraint on the mean losses beyond that α^{th} percentile is almost similar, except for a rare case when there is a high probability atom sitting beyond the α^{th} percentile. This is because once a risk threshold is crossed, there is very little that can be done to avoid its effects, e.g. once a certain flood limit is reached in a river, the damage is done. Therefore, we use VaR as a risk indicator in this thesis. Also, VaR can be easily accommodated in the intended M-CCP formulation, as described earlier.

4.1.3 Stochastic MPC based optimisation problem

In this section we state the type of optimisation problem (a Multiple Chance-Constrained optimisation Problem (M-CCP)) that we intend to solve in a receding horizon fashion in a Stochastic MPC setting.

The states representation of the system \mathcal{S} (Eqs. 4.1 & 4.2) over a finite horizon M is given below (for details, see Section 3.2.1),

$$\mathbf{x} = Fx_n + G\mathbf{u} + H\mathbf{w}, \quad (4.9)$$

with

$$F = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^M \end{bmatrix}, \quad G = \begin{bmatrix} B & \mathbf{0} & \dots & \mathbf{0} \\ AB & B & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ A^{M-1}B & A^{M-2}B & \dots & B \end{bmatrix}, \quad H = \begin{bmatrix} I & \mathbf{0} & \dots & \mathbf{0} \\ A & I & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ A^{M-1} & A^{M-2} & \dots & I \end{bmatrix},$$

where $\mathbf{x} = [x_{n+1} \ x_{n+2} \ \dots \ x_{n+M}]^\top$, $\mathbf{u} = [u_n \ u_{n+1} \ \dots \ u_{n+M-1}]^\top$, $\mathbf{w} = [w_n \ w_{n+1} \ \dots \ w_{n+M-1}]^\top$, and $\mathbf{w} \in \mathbb{W} = W^M$. For notational simplicity, we do not write time indices with these vectors in this chapter. Furthermore, as mentioned earlier, we assume the disturbances w_n, w_{n+1}, \dots are independent and identically distributed.

Next, we first introduce the two chance-constraints, i.e. the one related to normal operations and the other related to risk mitigation, and then we present the M-CCP problem that we aim to solve.

Chance-constraint related to normal operations: We consider that the following chance-constraint is related to the normal control operations of the system S ,

$$\mathbb{P}\{\mathbf{w} \in \mathbb{W} : \mathbf{u} \in \mathcal{U} \cap \bar{f}(\mathbf{u}, \mathbf{w}) \leq 0\} \geq 1 - \epsilon, \quad (4.10)$$

where $\mathbb{P} = \mathbb{P}_W^M$, $\mathcal{U} = U^M$ and $\bar{f}(\mathbf{u}, \mathbf{w})$ denotes the infinity norm on a vector of functions $(\mathbf{f}(\mathbf{u}, \mathbf{w}))$ defined over the finite time horizon M , i.e. $\bar{f}(\mathbf{u}, \mathbf{w}) = \|\mathbf{f}(\mathbf{u}, \mathbf{w})\|_\infty$, where $\mathbf{f}(\mathbf{u}, \mathbf{w})$ is given by,

$$\begin{aligned} \mathbf{f}(\mathbf{u}, \mathbf{w}) = & [f_n(u_n, w_n) - \chi_n \quad f_{n+1}(u_{n+1}, w_{n+1}) - \chi_{n+1} \quad \dots \\ & \dots \quad f_{n+M-1}(u_{n+M-1}, w_{n+M-1}) - \chi_{n+M-1}]^\top, \end{aligned} \quad (4.11)$$

χ_i is an upper limit on a function $f_i(u_i, w_i)$ (related to the normal control operations, which depends on the input variables u_i and uncertain variables w_i), i.e. ' $f_i(u_i, w_i) - \chi_i \leq 0$ ' or ' $f_i(u_i, w_i) \leq \chi_i$ ', where $i = n, n+1, \dots, n+M-1$, and i denotes the time index over the finite horizon M . Therefore, $\bar{f}(\mathbf{u}, \mathbf{w})$ is the maximum deviation of the function $f(u, w)$ from its limits over the finite horizon. $\epsilon \in (0, 1)$ in Eq. 4.10 is the allowed violation probability.

Chance-constraint related to risk mitigation: We consider that the following chance-constraint is related to the risk mitigation operations of the system S ,

$$\mathbb{P}\{\mathbf{w} \in \mathbb{W} : \bar{g}(\mathbf{u}, \mathbf{w}) \leq 0\} \geq 1 - \epsilon_V, \quad (4.12)$$

where $\bar{g}(\mathbf{u}, \mathbf{w})$ denotes the infinity norm on a vector of functions $(\mathbf{g}(\mathbf{u}, \mathbf{w}))$ defined over

the finite time horizon M , i.e. $\bar{g}(\mathbf{u}, \mathbf{w}) = \|\mathbf{g}(\mathbf{u}, \mathbf{w})\|_\infty$, where $\mathbf{g}(\mathbf{u}, \mathbf{w})$ is given by,

$$\mathbf{g}(\mathbf{u}, \mathbf{w}) = [g_n(u_n, w_n) - \gamma_n \quad g_{n+1}(u_{n+1}, w_{n+1}) - \gamma_{n+1} \quad \dots \\ \dots \quad g_{n+M-1}(u_{n+M-1}, w_{n+M-1}) - \gamma_{n+M-1}]^\top, \quad (4.13)$$

$g(u, x)$ is the loss function associated with system risks (see Section 4.1.1), γ_i is an upper limit on the function $g_i(u_i, w_i)$ (see Eq. 4.7), i.e. ' $g_i(u_i, w_i) - \gamma_i \leq 0$ ' or ' $g_i(u_i, w_i) \leq \gamma_i$ ', where $i = n, n+1, \dots, n+M-1$. Therefore, $\bar{g}(\mathbf{u}, \mathbf{w})$ is the maximum deviation of the function $g(u, w)$ from its limits over the finite horizon M . $\epsilon_V \in (0, 1)$ in Eq. 4.12 is the allowed violation probability, where $1 - \epsilon_V = \alpha$ (from Eq. 4.7). We replaced α with $1 - \epsilon_V$ in Eq. 4.7 to maintain notational consistency between the two chance-constraints (Eqs. 4.10 & 4.12). Also, based on the priority given to the satisfaction of the risk mitigation operations, we assume $\epsilon_V \ll \epsilon$.

To state the optimisation problem for Stochastic MPC, we use Eq. 3.52 as the objective function $J(\mathbf{u}, \mathbf{w})$ of the normal control operations, which includes deviation of states \mathbf{x} from a reference \mathbf{x}_r , inputs \mathbf{u} and change in the inputs. Eq. 3.52 and the constraints in Eqs. 4.10 & 4.12 form the following M-CCP, which can be solved in a Stochastic MPC setting,

$$\begin{aligned} & \min_{\Gamma, \Theta} J(\mathbf{u}, \mathbf{w}), & (4.14) \\ \text{s.t.} & \quad \mathbb{P}\{\mathbf{w} \in \mathbb{W} : \mathbf{u}(\mathbf{w}) \in \mathcal{U} \cap \bar{f}(\mathbf{u}, \mathbf{w}) \leq 0\} \geq 1 - \epsilon, \\ & \quad \mathbb{P}\{\mathbf{w} \in \mathbb{W} : \bar{g}(\mathbf{u}, \mathbf{w}) \leq 0\} \geq 1 - \epsilon_V, \end{aligned}$$

subject to the states dynamics, \mathbf{x} in Eq. 4.9, and the parametrisation of control policies \mathbf{u} in \mathbf{w} , e.g. an affine parametrisation in \mathbf{w} , as stated ahead in Eq. 4.15. We intend to incorporate the states dynamics in the objective function $J(\mathbf{u}, \mathbf{w})$ (e.g. see Eqs. 3.52 & 3.53). Problem (4.14) is an M-CCP with two chance-constraints, and we make the following assumptions related to this problem,

- \mathcal{U} and \mathbb{W} are convex sets¹.

¹A set \mathcal{X} is said to be convex, if for all $x_1, x_2 \in \mathcal{X}$ and $\alpha \in [0, 1]$, then the point $\alpha x_1 + (1 - \alpha)x_2$ also belongs to \mathcal{X} .

- The functions $\bar{f}(\mathbf{u}, \mathbf{w})$, $\bar{g}(\mathbf{u}, \mathbf{w})$ and $J(\mathbf{u}, \mathbf{w})$ are convex² with respect to \mathbf{u} , where \mathbf{u} is assumed to be a vector of control policies parametrised in \mathbf{w} , and we use the following affine parametrisation of the control policies (as in Eq. 3.16),

$$\mathbf{u} = \Theta \mathbf{w} + \Gamma, \quad (4.15)$$

where Θ and Γ are the parameter matrices (for details see Section 3.2.2), and they are the optimisation variables of Problem (4.14). In this chapter, we denote the solutions of problems, derived from Problem (4.14), as e.g. \mathbf{u}^* and $\bar{\mathbf{u}}^*$, which will be the values of the vector \mathbf{u} , obtained from Eq. 4.15 using different Θ and Γ matrices.

We also assumed that the first chance-constraint of Problem (4.14) is associated with normal control operations and the second chance-constraint is associated with risk mitigation. However, this categorisation into normal operations and risk mitigation is not necessary. The proposed algorithm in this chapter is valid for solving any M-CCP with two chance-constraints with the allowed violation probabilities satisfying $\epsilon_V < \epsilon$, however, it is developed specifically for the case $\epsilon_V \ll \epsilon$. The proposed algorithm is also generalisable to an M-CCP with any finite number of chance constraints, which will be discussed in Section 4.5.

4.2 Randomised approaches to solve an M-CCP with two chance-constraints

In this section we first discuss some alternative ways, available in the literature, to find an approximate solution to Problem (4.14), and then we present an intuitive idea behind the proposed *Optimisation and Testing algorithm* to solve the problem. The algorithm is formally presented in the next section.

4.2.1 Some possible ways to solve Problem (4.14)

There are several randomised ways to solve Problem (4.14),

²A function $f : \mathcal{X} \rightarrow \mathbb{R}$ is a convex function, if for all $x_1, x_2 \in \mathcal{X}$ and $\alpha \in [0, 1]$, then $f(\alpha x_1 + (1 - \alpha)x_2) \leq \alpha f(x_1) + (1 - \alpha)f(x_2)$.

- A feasible solution to a chance-constrained problem can be found by resorting to the scenario approach, see Section 3.2.4. A naive application of the scenario approach to the multiple chance-constrained Problem (4.14) is to set $\epsilon = \epsilon_V$. However, the small value of $\epsilon_V \ll \epsilon$, would require a very large number of constraints in the scenario problem according to Eq. 3.24. The computational effort is therefore large, and thus this is not a viable option in general.
- There are some recent results on the scenario approach presented in [122], that target specifically multiple chance-constrained problems. The results in [122] suggests to apply the scenario approach individually to each chance-constraint. Those results improve on the naive approach only if the chance-constraints are sufficiently decoupled, i.e. each chance-constraint involves different decision variables. However, still a small value of ϵ_V in the second chance-constraint would lead to a large computational effort in solving the corresponding scenario problem.
- In [127], we proposed an iterative optimisation and testing algorithm to find approximate solutions to M-CCPs. The algorithm uses the scenario approach together with ideas borrowed from validation set methods [118, 123–125] to provide computationally tractable solutions to M-CCPs. It performs an optimisation with the chance-constraint corresponding to normal operations only, and then test the solution against the risk mitigation chance-constraint. The test decides whether an improvement is required or not. However, the selection of certain parameters in the improving phase, including the number of iterations are based on heuristics. Moreover, the satisfaction of the required probabilistic guarantees can demand many iterations, which can be computationally expensive.

In this chapter we propose an improved non-iterative *Optimisation and Testing algorithm*, and remove the drawbacks in our previous proposal [127]. The algorithm provides a solution, if it exists, in a fixed and finite number of steps with rigorous probabilistic guarantees. The algorithm is formally described in the next section. In the next subsection, we provide the basic idea. However, before that, we first introduce the concept of a *default solution* which is used in the algorithm.

Default solution

A *default solution* refers to a possible solution which can be used, e.g. when a given or found solution does not satisfy the required constraints. The *default solution* does not necessarily have to be an outcome of an optimisation problem. It can possibly belong to a set larger than \mathcal{U} (in Problem (4.14)), e.g. it can belong to a set \mathcal{U}' , where $\mathcal{U} \subseteq \mathcal{U}'$. The idea is to have a backup solution, which in terms of objective criterion may give a poor performance, but is the safest option when the situation takes a turn for the worse. E.g. in case of river flooding, the solution might refer to shutting or opening the storage gates completely. This is how river operators proceed when necessary. Different applications will have different types of *default solution*. We assume that such a solution exists, and denote it as \mathbf{u}_0^* .

4.2.2 Intuitive description of the *Optimisation and Testing algorithm*

The *Optimisation and Testing algorithm* has three main steps,

1. **Optimisation:** We first solve an optimisation problem with only the first chance-constraint (the chance-constraint related to normal operations) in Problem (4.14), i.e.

$$\begin{aligned} & \min_{\Gamma, \Theta} J(\mathbf{u}, \mathbf{w}), & (4.16) \\ \text{s.t.} \quad & \mathbb{P}\{\mathbf{w} \in \mathbb{W} : \mathbf{u}(\mathbf{w}) \in \mathcal{U} \cap \bar{f}(\mathbf{u}, \mathbf{w}) \leq 0\} \geq 1 - \epsilon. \end{aligned}$$

To find an approximate solution of Problem (4.16), we employ the scenario approach (Theorem 3.1). We generate N_r independent realisations of the disturbance \mathbf{w} , according to a given probability distribution (Eq. 3.24), and replace Problem (4.16) with the following scenario problem,

$$\begin{aligned} & \min_{\Gamma, \Theta} J(\mathbf{u}, \mathbf{w}), & (4.17) \\ \text{s.t.} \quad & \mathbf{u}(\mathbf{w}^{(k)}) \in \mathcal{U}, \quad \bar{f}(\mathbf{u}, \mathbf{w}^{(k)}) \leq 0, \\ & \text{for } k = 1, \dots, N_r. \end{aligned}$$

We solve Problem (4.17) and find a scenario solution, say \mathbf{u}^* .

2. **Testing:** In the second step, we test the solution \mathbf{u}^* of Problem (4.17) against the second chance-constraint (the chance-constraint related to risk mitigation) in Problem (4.14), by resorting to a Monte-Carlo sample of N_T new scenarios. This is computationally cheap since no optimisation is performed. If the number of scenarios violating the constraint is below a selected threshold, we use the solution (\mathbf{u}^*) and provide a certificate with a probabilistic guarantee that the solution satisfies both chance-constraints. Otherwise, we improve the obtained solution.
3. **Improving:** If the scenario solution \mathbf{u}^* fails the aforementioned test, we test the *default solution* ($\mathbf{u}_0^* \in \mathcal{U}'$), against the second chance-constraint for the same N_T scenarios. If the violations are less than the selected threshold, then we solve a one-dimensional scenario problem, and improve feasibility of the scenario solution \mathbf{u}^* , by moving it in the direction of the default solution \mathbf{u}_0^* along the line $(1 - \alpha)\mathbf{u}^* + \alpha\mathbf{u}_0^*$, where $\alpha \in (0, 1]$. In the scenario problem, we minimise the value of α , and consider the scenarios $\mathbf{w} \in \mathbb{W}$ that violate the risk-mitigation (second) constraint with \mathbf{u}^* to ensure improvement, but satisfy the constraint with \mathbf{u}_0^* to ensure the feasibility of the scenario problem. Moreover, we relax the constraint on the control action \mathbf{u} in the improvement phase. However, if the *default solution* fails the test against the second chance-constraint, then we exit the algorithm and inform system operators, which initiates emergency operations. E.g. for rivers, flood operations can be pursued at such an occasion. The emergency operations are outside the scope of this thesis.

N_r and N_T are computed in such a way that the solution obtained from the algorithm comes with precise probabilistic guarantees. In particular, if the algorithm successfully ends after the testing procedure, a certificate is delivered that states that both the chance-constraints in Problem (4.14) are satisfied with high confidence. Otherwise, the first chance-constraint (of Problem (4.14)) can be tested with the improved solution, and depending on that, a certificate with the satisfaction of both chance-constraints or a certificate guaranteeing only the second chance-constraint is delivered. Fig. 4.4 summarises

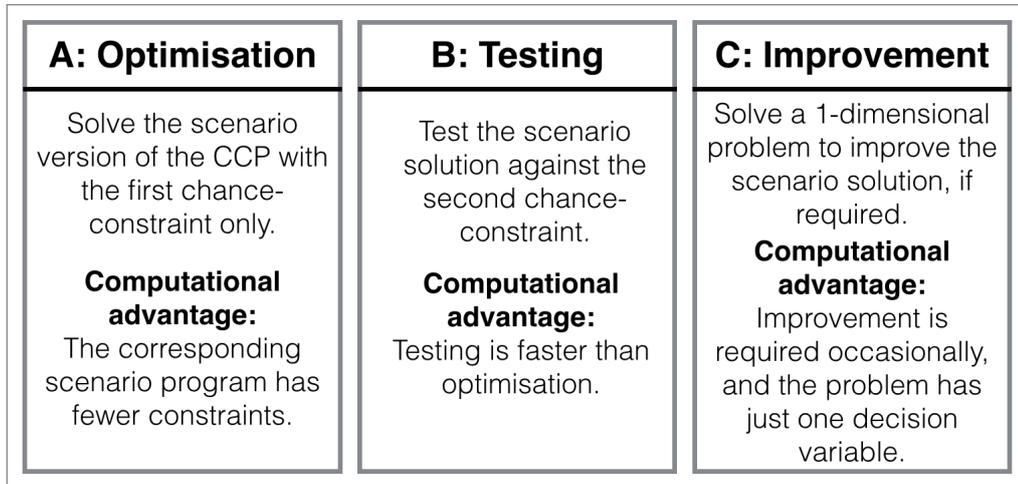


Figure 4.4: *Optimisation and Testing algorithm* (basic idea).

the idea of the algorithm with an emphasis on the computational advantages that it provides in comparison to the alternative approaches mentioned above.

4.3 Optimisation and Testing algorithm to solve an M-CCP with 2 chance-constraints

In this section we provide technical explanation of various steps in the algorithm and state auxiliary procedures and terms, which we later use to formally state the *Optimisation and Testing algorithm*.

4.3.1 Auxiliary terms and procedures

For each procedure, we describe the need and the background of that procedure, before formally stating it in *italics*.

1. **Number of Scenarios (N_r) required to find an approximate solution of a chance-constrained (CC) problem; *Find_N_CC*(ϵ, β, d):**

We use the scenario approach (Section 3.2.4) to get an approximate solution of a chance-constrained problem. This procedure is defined to find the number of scenarios N_r required in the scenario problem, whose solution will be feasible for the

original chance-constrained problem with a confidence $1 - \beta$, where $\beta \in (0, 1)$ is a confidence parameter.

The procedure $\text{Find_N_CC}(\epsilon, \beta, d)$ takes in the desired violation probability $\epsilon \in (0, 1)$ (of the chance-constraint), the desired confidence parameter $\beta \in (0, 1)$ and the number of decision variables d , and uses Eq. 3.24 to find the required minimum number of scenarios N_r for the scenario problem.

In practice, $\text{Find_N_CC}(\epsilon, \beta, d)$ can be computed easily, see e.g. the bisection algorithms in [128, 129].

2. Number of Scenarios (N_T) required to test a solution against a chance-constraint that must be satisfied with probability $1 - \epsilon_V$; $\text{Find_N_Test}(\epsilon_V, \rho, \beta_V)$:

As mentioned in Section 4.2.2, we find a scenario solution, \mathbf{u}^* , to Problem (4.17), and then we test the solution against the second chance-constraint (the chance-constraint related to risk mitigation) in Problem (4.14). Here, we first describe the background of the *testing* procedure, then present a mechanism to determine number of scenarios N_T required for *testing*, and then state the procedure formally

Background of the *testing* procedure:

We test the solution \mathbf{u}^* against the chance-constraint $\mathbb{P}\{\mathbf{w} \in \mathbb{W} : \bar{g}(\mathbf{u}^*, \mathbf{w}) \leq 0\} \geq 1 - \epsilon_V'$ in a randomised way, by doing N_T Bernoulli trials. In a Bernoulli trial, we evaluate the constraint $\bar{g}(\mathbf{u}^*, \mathbf{w}^{(k)}) \leq 0$ with an independently drawn realisation $\mathbf{w}^{(k)} \in \mathbb{W}$. If the constraint is violated, the corresponding Bernoulli random variable, say B_k , takes the value equal to 1, and if the constraint is satisfied, the corresponding B_k takes the value equal to 0, i.e.

$$B_k = \mathbb{1}\{\bar{g}(\mathbf{u}^*, \mathbf{w}^{(k)}) > 0\}, \quad \text{for } k = 1, 2, \dots, N_T,$$

where $\mathbb{1}\{\cdot\}$ denotes the indicator function. The probability (p) of the variable B_k to be 1 is: $\mathbb{P}(B_k = 1) = \mathbb{P}\{\mathbf{w}^{(k)} \in \mathbb{W} : \bar{g}(\mathbf{u}^*, \mathbf{w}^{(k)}) > 0\}$, i.e., it is equal to the violation probability of the constraint $\bar{g}(\mathbf{u}^*, \mathbf{w}) \leq 0$ for a realisation of $\mathbf{w} \in \mathbb{W}$, which is what we want to estimate. An empirical estimate of the required violation probability,

say \hat{p} , can be obtained as,

$$\hat{p} = \frac{1}{N_T} \sum_{k=1}^{N_T} B_k = \frac{1}{N_T} \sum_{k=1}^{N_T} \mathbb{1}\{\bar{g}(\mathbf{u}^*, \mathbf{w}^{(k)}) > 0\}. \quad (4.18)$$

Eq. 4.18 states \hat{p} as the fraction of the drawn realisations of $\mathbf{w} \in \mathbb{W}$, that violate the constraint ' $\bar{g}(\mathbf{u}^*, \mathbf{w}) \leq 0$ '. After calculating \hat{p} , we compare it with the allowed violation probability ϵ_V . We declare the test to be passed and the chance-constraint to be satisfied, if $\hat{p} \leq \epsilon_V - \varrho$, where $\varrho \in (0, \epsilon_V)$ is a safety margin (e.g. $\varrho = \frac{1}{2}\epsilon_V$). Note that the total number of violations, say v , in N_T Bernoulli trials is a Binomial random variable. We denote the random variable as V , and the probability of $V = v$ is

$$\mathbb{P}(V = v, p) = \binom{N_T}{v} p^v (1-p)^{N_T-v}, \quad (4.19)$$

where the above probability is a function of the actual (unknown) violation probability p .

How to determine number of scenarios N_T :

For the *testing* procedure described above, we need a mechanism to find the number N_T and to provide probabilistic guarantees on the test, and for that we proceed as follows. We bound the probability of passing the test (i.e. $\hat{p} \leq \epsilon_V - \varrho$), when the actual probability $p > \epsilon_V$, by a user chosen confidence parameter $\beta_V \in (0, 1)$, i.e.

$$\mathbb{P}(\hat{p} \leq \epsilon_V - \varrho | p > \epsilon_V) \leq \beta_V, \quad (4.20)$$

where β_V should be ideally very small (e.g. 10^{-6}), and \mathbb{P} is a measure on the set \mathbb{W} to which \mathbf{w} belongs. The expression $\mathbb{P}(\hat{p} \leq \epsilon_V - \varrho)$ can be obtained by using the Binomial Cumulative Distribution Function (Binom-CDF), because it is equivalent to check the cumulative probability of the Binomial random variable, $V = N_T \hat{p}$, to stay below the required number of violations, $N_T(\epsilon_V - \varrho)$. For a given $p = \mu$, we get

the following expression from the definition of Binom-CDF,

$$\mathbb{P}(\hat{p} \leq \epsilon_V - \varrho | p = \mu) = \sum_{i=0}^{\lfloor N_T(\epsilon_V - \varrho) \rfloor} \binom{N_T}{i} \mu^i (1 - \mu)^{N_T - i}, \quad (4.21)$$

i.e. we add the probabilities of all Binomial random variables (Eq. 4.19) with violations less than or equal to $N_T(\epsilon_V - \varrho)$, and we can further say

$$\begin{aligned} \mathbb{P}(\hat{p} \leq \epsilon_V - \varrho | p = \mu > \epsilon_V) &\leq \sup_{\mu > \epsilon_V} \sum_{i=0}^{\lfloor N_T(\epsilon_V - \varrho) \rfloor} \binom{N_T}{i} \mu^i (1 - \mu)^{N_T - i}, \\ &= \sum_{i=0}^{\lfloor N_T(\epsilon_V - \varrho) \rfloor} \binom{N_T}{i} \epsilon_V^i (1 - \epsilon_V)^{N_T - i}. \end{aligned}$$

The last equality holds true because the expression on the right hand side of the equation is monotonically decreasing with respect to μ , and $\mu = \epsilon_V$ provides the supremum. As stated earlier, we want to bound this expression by β_V (Eq. 4.20) i.e.

$$\mathbb{P}(\hat{p} \leq \epsilon_V - \varrho | p > \epsilon_V) \leq \sum_{i=0}^{\lfloor N_T(\epsilon_V - \varrho) \rfloor} \binom{N_T}{i} \epsilon_V^i (1 - \epsilon_V)^{N_T - i} \leq \beta_V. \quad (4.22)$$

We use the above expression to find the minimum number of scenarios N_T required to test the second chance-constraint in Problem (4.14), which can be violated with probability at most $\epsilon_V - \varrho$, with confidence at least $1 - \beta_V$, i.e.

$$\min\{N_T \in \mathbb{N} : \sum_{i=0}^{\lfloor N_T(\epsilon_V - \varrho) \rfloor} \binom{N_T}{i} \epsilon_V^i (1 - \epsilon_V)^{N_T - i} \leq \beta_V\}, \quad (4.23)$$

where \mathbb{N} is a set of natural numbers.

The procedure $\text{Find_N_Test}(\epsilon_V, \varrho, \beta_V)$ takes in the allowed violation probability $\epsilon_V \in (0, 1)$, a suitable additive margin $\varrho \in (0, \epsilon_V)$ and the desired confidence parameter $\beta_V \in (0, 1)$, and uses Eq. 4.23 to provide the minimum number of scenarios N_T , required to test a chance-constraint, based on the selected parameters, in a randomised way.

The above results are also concluded in the following theorem,

Theorem 4.1. *The number of scenarios (N_T) of the uncertain variable $\omega \in \mathbb{W}$ required*

4.3 Optimisation and Testing algorithm to solve an M-CCP with 2 chance-constraints 67

to certify the validity of a \mathbf{w} -dependent constraint, that can be violated with probability at most $\epsilon_V \in (0, 1)$, with confidence $1 - \beta_V$, can be determined by the following expression

$$\min\{N_T \in \mathbb{N} : \sum_{i=0}^{\lfloor N_T(\epsilon_V - \varrho) \rfloor} \binom{N_T}{i} \epsilon_V^i (1 - \epsilon_V)^{N_T - i} \leq \beta_V\}, \quad (4.24)$$

where $\beta_V \in (0, 1)$ is a confidence parameter, $\varrho < \epsilon_V$ is a safety margin and \mathbb{N} is a set of natural numbers.

3. Upper bound ($\hat{\epsilon}$) on a violation probability p , based on its estimate \hat{p} ; Upper Bound($\hat{p}, \beta_{\hat{\epsilon}}, N_T$):

In the proposed algorithm (in Section 4.3.2), we need to calculate upper bounds on violation probabilities (e.g. p in the previous procedure). Here, we first provide the background and a technical discussion that will lead to this requirement, before describing a procedure to calculate an upper bound and stating it formally.

Background:

If the solution, \mathbf{u}^* , of Problem (4.17) does not satisfy the second chance-constraint in Problem (4.14), based on the test described in the previous procedure (Find_N_Test($\epsilon_V, \varrho, \beta_V$)), then we test the *default solution*, \mathbf{u}_0^* , against the second chance-constraint, by using the procedure Find_N_Test($\epsilon_V, \varrho, \beta_V$). Let \hat{p}_0 denote the empirical estimate of the violation probability of \mathbf{u}_0^* (similar to Eq. 4.18),

$$\hat{p}_0 = \frac{1}{N_T} \sum_{k=1}^{N_T} \mathbb{1}\{\bar{g}(\mathbf{u}_0^*, \mathbf{w}^{(k)}) > 0\}, \quad (4.25)$$

If $\hat{p}_0 \leq \epsilon_V - \varrho$, i.e. the above mentioned test passes, and the *default solution* satisfies the second chance-constraint, then we improve the feasibility of the scenario solution \mathbf{u}^* , by solving a one-dimensional scenario problem. In the scenario problem, we move \mathbf{u}^* in the direction of the default solution, \mathbf{u}_0^* , along the line $(1 - \alpha)\mathbf{u}^* + \alpha\mathbf{u}_0^*$, while minimising the value of α , as follows,

$$\min_{\alpha} J(\bar{\mathbf{u}}, \mathbf{w}), \quad (4.26)$$

$$\begin{aligned} \text{s.t. } \quad & \bar{\mathbf{u}}(\mathbf{w}^{(k)}) \in \mathcal{U}', \quad \bar{g}(\bar{\mathbf{u}}, \mathbf{w}^{(k)}) \leq 0, \\ & \text{for } k = 1, 2, \dots, N_\alpha, \quad \mathbf{w}^{(k)} \in \mathcal{T}, \end{aligned}$$

where $\alpha \in (0, 1]$ and $\bar{\mathbf{u}} = (1 - \alpha)\mathbf{u}^* + \alpha\mathbf{u}_0^*$. In the above problem we minimise the value of α , because we want to stay as close as possible to the scenario solution \mathbf{u}^* . If α^* is a solution of Problem (4.26), then the improved solution is $\bar{\mathbf{u}}^* = (1 - \alpha^*)\mathbf{u}^* + \alpha^*\mathbf{u}_0^*$. Also, note that while *improving*, we only include the constraint related to risk mitigation. The set \mathcal{T} and the number of scenarios N_α , used in Problem (4.26), are discussed next. We define

$$\mathcal{T} = \{\mathbf{w} \in \mathbb{W} \mid \bar{g}(\mathbf{u}^*, \mathbf{w}) > 0 \wedge \bar{g}(\mathbf{u}_0^*, \mathbf{w}) \leq 0\} \subseteq \mathbb{W}, \quad (4.27)$$

which is important for improvement purposes. It contains realisations of $\mathbf{w} \in \mathbb{W}$, that satisfy the second chance-constraint with \mathbf{u}_0^* to ensure the feasibility of Problem (4.26), but does not satisfy the second chance-constraint with \mathbf{u}^* to ensure improvement. The practice of sampling \mathbf{w} from a set of interest (e.g. $\mathcal{T} \subseteq \mathbb{W}$, with small probability) bears similarities with ‘importance sampling’ (see e.g. [130]). In our case it is easy to find such a set, because we generate N_T scenarios of \mathbf{w} and perform tests on the two solutions before solving Problem (4.26), so during the *testing* procedure we can construct a set, say $Q \subseteq \mathcal{T}$, by saving all such scenarios in that set.

The number of scenarios N_α is calculated by calling the procedure `Find_N_CC`($\epsilon_\alpha, \beta_\alpha, 1$), where β_α is a confidence parameter, ideally very small (e.g. 10^{-6}) and ϵ_α is the allowed (conditional) violation probability for the chance-constrained problem corresponding to the scenario Problem (4.26). We added the word ‘conditional’, because ϵ_α is the violation probability, conditioned on the set \mathcal{T} inside \mathbb{W} . It is calculated by the following expression,

$$\epsilon_\alpha = \frac{\epsilon_V - \hat{\epsilon}_0}{\hat{\epsilon}_T}, \quad (4.28)$$

where $\hat{\epsilon}_0$ is an upper bound on the probability of a $\mathbf{w} \in \mathbb{W}$ realisation to violate the

the second chance-constraint in Problem (4.14) with \mathbf{u}_0^* , and $\hat{\epsilon}_T$ is an upper bound on the probability of a $\mathbf{w} \in \mathbb{W}$ realisation to violate the chance-constraint with \mathbf{u}^* only. We describe a procedure to calculate these upper bounds ahead. Eq. 4.28 is obtained from Theorem 4.2 stated below. Let $VS(\mathbf{u}_0^*)$ be the set, $VS(\mathbf{u}_0^*) = \{\mathbf{w} \in \mathbb{W} \mid \bar{g}(\mathbf{u}_0^*, \mathbf{w}) > 0\}$.

Theorem 4.2. *If $\mathbb{P}(\mathcal{T}) \leq \hat{\epsilon}_T$ and $\mathbb{P}(VS(\mathbf{u}_0^*)) \leq \hat{\epsilon}_0$, and if the number of scenarios N_α for Problem (4.26) is computed from `Find_N_Scenario`($\epsilon_\alpha, \beta_\alpha, 1$), with*

$$\epsilon_\alpha \leq \frac{\epsilon_V - \hat{\epsilon}_0}{\hat{\epsilon}_T}, \quad (4.29)$$

and the scenarios are sampled from the set \mathcal{T} , then the control action $\bar{\mathbf{u}}^ = (1 - \alpha^*)\mathbf{u}^* + \alpha^*\mathbf{u}_0^*$, obtained from the solution of Problem (4.26), violates the constraint ' $\bar{g}(\bar{\mathbf{u}}^*, \mathbf{w}) \leq 0$ ' with a probability, no more than ϵ_V , with a confidence $1 - \beta_\alpha$.*

A proof of this theorem, with its explanation, is in Appendix 4.A. Eq. 4.29 supports the fact that the larger the violation bound $\hat{\epsilon}_0$ of the *default solution* is, the smaller will be the allowed violation probability ϵ_α , and in turn it will require a higher number of scenarios N_α for Problem (4.26) and vice versa. Similarly, the larger the violation bound $\hat{\epsilon}_T$ of the scenario solution \mathbf{u}^* is, the smaller ϵ_α will be, and in turn it will also require a higher number of scenarios N_α for Problem (4.26) and vice versa. Next, we describe a way to find an upper bound on the violation probabilities discussed here and in the previous procedure.

A procedure to find an upper bound on a violation probability:

An upper bound on a violation probability, e.g. p , can be obtained from the *testing* procedure (`Find_N_Test`($\epsilon_V, \varrho, \beta_V$)) as $\hat{p} + \varrho$, because there we ensure $\hat{p} \leq \epsilon_V - \varrho$ with confidence $1 - \beta_V$, where ϱ is a safety margin. This can be conservative at times, especially, when we have very small probability values, because we will always add the fixed ϱ (e.g. $\epsilon_V/2$) to the estimate \hat{p} , to find an upper bound. Such relatively large upper bounds will give smaller ϵ_α in Eq. 4.28, which will cost more computational power to solve Problem (4.26). Therefore, we use the following alternative approach, that uses the estimate \hat{p} and the number of scenarios used in *testing*, N_T ,

Violations v	Probabilities $\mathbb{P}(V = v, p)$	Empirical estimate of p (\hat{p})
0	$\mathbb{P}(V = 0, p) = \binom{N_T}{0} p^0 (1-p)^{N_T}$	$\frac{0}{N_T}$
1	$\mathbb{P}(V = 1, p) = \binom{N_T}{1} p^1 (1-p)^{N_T-1}$	$\frac{1}{N_T}$
2	$\mathbb{P}(V = 2, p) = \binom{N_T}{2} p^2 (1-p)^{N_T-2}$	$\frac{2}{N_T}$
.	.	.
.	.	.
$N_T - 1$	$\mathbb{P}(V = N_T - 1, p) = \binom{N_T}{N_T-1} p^{N_T-1} (1-p)^1$	$\frac{N_T-1}{N_T}$
N_T	$\mathbb{P}(V = N_T, p) = \binom{N_T}{N_T} p^{N_T} (1-p)^0$	$\frac{N_T}{N_T}$

Table 4.1: Binomial random variable V with its possible outcomes and probabilities.

and provide an improved upper bound, $\hat{\epsilon}$, compared to $\hat{p} + \varrho$, again with the help of Binomial Cumulative Distribution Function (Binom-CDF).

In the *testing* procedure, we test the solution, \mathbf{u}^* , against the constraint $\bar{g}(\mathbf{u}^*, \mathbf{w}^{(k)}) \leq 0$, for $k = 1, 2, \dots, N_T$. As described in the previous procedure, the number of violations, v , in the N_T Bernoulli trials is a Binomial random variable, V , where $v \in \{0, 1, 2, \dots, N_T\}$. The random variable V can pick one of the $N_T + 1$ outcomes in Table 4.1, and the associated probability, $\mathbb{P}(V = v, p)$, is a function of the actual (unknown) probability of violation, p (Eq. 4.19).

The Binom-CDF can be used to find the upper bound $\hat{\epsilon}$, with the available information of the number of violations v , number of Bernoulli trials N_T and a user selected confidence parameter $\beta_{\hat{\epsilon}}$, which is ideally very small (e.g. 10^{-6}). We define the Binom-CDF as below (see Table 4.1),

$$\mathbb{F}\left(\frac{v}{N_T}, p\right) = \mathbb{P}\left(\hat{p} \leq \frac{v}{N_T}, p\right) = \sum_{i=0}^v \mathbb{P}(V = i, p).$$

The function $\mathbb{F}\left(\frac{v}{N_T}, p\right)$ is non-increasing with respect to p . It is equal to 1, when p is equal to 0, for any value of v , and it is equal to 0, when p is equal to 1, for any value of v except when $v = N_T$, and in that case the function $\mathbb{F}\left(\frac{v}{N_T}, p\right)$ is equal to 1 for all values of p . Our requirement is to keep $\mathbb{F}\left(\frac{v}{N_T}, p\right) = \mathbb{P}\left(\hat{p} \leq \frac{v}{N_T}, p\right) \geq \beta_{\hat{\epsilon}}$, and we want to include all values on the p axis in an interval where $\mathbb{P}\left(\hat{p} \leq \frac{v}{N_T}, p\right) \geq \beta_{\hat{\epsilon}}$, and the right limit of that interval will provide the upper bound. If we plot the function

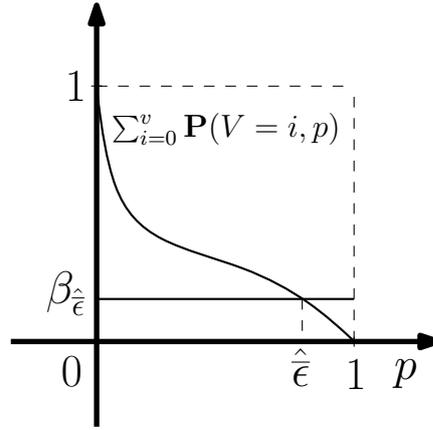


Figure 4.5: A plot of $\mathbb{F}(\frac{v}{N_T}, p)$ against p .

$\mathbb{F}(\frac{v}{N_T}, p)$ against p (for an example, see the plot in Fig. 4.5), then the value on the p axis, where the curve intersects $\beta_{\hat{\epsilon}}$ is the required upper bound, $\hat{\epsilon}$, on the actual violation probability. This is because up till this value on the p axis, the requirement $\sum_{i=0}^v \mathbb{P}(V = i, p) \geq \beta_{\hat{\epsilon}}$ is satisfied, and the curve $\sum_{i=0}^v \mathbb{P}(V = i, p)$ lies above $\beta_{\hat{\epsilon}}$ up till this value, as shown in Fig. 4.5.

Therefore, an upper bound $\hat{\epsilon}$ can be found by solving the following equation for p ,

$$\sum_{i=0}^v \mathbb{P}(V = i, p) = \sum_{i=0}^v \binom{N_T}{i} p^i (1-p)^{N_T-i} = \beta_{\hat{\epsilon}}, \quad (4.30)$$

and then assigning the solution to $\hat{\epsilon}$. The actual probability p will lie in the interval $[0, \hat{\epsilon}]$, with confidence $1 - \beta_{\hat{\epsilon}}$.

The procedure `Upper_Bound($\hat{p}, \beta_{\hat{\epsilon}}, N_T$)` takes in the violation probability estimate ($\hat{p} = v/N_T$) and the number of scenarios (N_T) used to find \hat{p} . It solves Eq. 4.30 for p and returns the upper bound $\hat{\epsilon}$ on the actual violation probability with confidence $1 - \beta_{\hat{\epsilon}}$.

The bound $\hat{\epsilon}$ can be found by using Matlab 'binofit' function as follows,

$$[\hat{p}, \text{Interval}] = \text{binofit}(v, N_T, 2\beta_{\hat{\epsilon}}),$$

where `Interval` = [lower bound, $\hat{\epsilon}$]. There are two outputs of the Matlab function: the first output is the estimated probability $\hat{p} = v/N_T$, and the second output is the

interval where the actual probability p lies. The interval also provides the lower bound, but we do not need it for our work. The first two inputs of the function are the number of violations v and the number of scenarios N_T used in the test. The third input ($2\beta_{\hat{\epsilon}}$) of the function is the user selected probability of the event where p lies outside the computed interval (Interval). It provides the confidence: $(1 - 2\beta_{\hat{\epsilon}}) \times 100\%$, that p lies in the computed interval. The function uses half of the probability $2\beta_{\hat{\epsilon}}$ in the calculation of the lower bound and half for the calculation of the upper bound. Although we are just interested in the upper bound, but to calculate that bound using the function `binofit`, we need to put $2\beta_{\hat{\epsilon}}$ as the third input, instead of $\beta_{\hat{\epsilon}}$. The literature provides a method that estimates the exact same interval as described above, using the Binomial distribution functions, which is called as Clopper-Pearson (CP) method. For details on the CP method, see [131] and the references therein.

4. Overall Probability Of Failure (POF):

δ is the overall violation probability or the overall POF of the algorithm. It is the probability of an event when the algorithm delivers a wrong certificate. It is set prior to the use of algorithm, and it should be kept very low, e.g. 10^{-8} . A wrong certificate can be issued, e.g. if a scenario solution of a scenario problem in the algorithm is not feasible for its corresponding chance-constrained optimisation problem, or when a test result is not correct, or when an upper bound obtained from the `Upper.Bound($\hat{p}, \beta_{\hat{\epsilon}}, N_T$)` procedure is not true.

4.3.2 Optimisation and Testing algorithm

As discussed in Section 4.2.2, the algorithm consists of three major steps, and we name them A, B & C. For each step, we first prescribe its sub-steps and then we add the details in remarks.

To begin with, select the margin $\varrho \in (0, \epsilon_V)$ (e.g. $\varrho = \epsilon_V/2$), and the overall POF (δ) of the optimisation algorithm (e.g. $\delta = 10^{-8}$).

Step A (Find a scenario solution u^* of Problem (4.16))

1. Use the violation probability ϵ (from Problem (4.16)) and the desired confidence parameter $\beta = \frac{\delta}{7}$ in the procedure $\text{Find_N_CC}(\epsilon, \beta, d)$, to determine the required number of scenarios N_r for the scenario Problem (4.17), where d is the number of optimisation variables in Problem (4.17).
2. Find the solution, u^* , of Problem (4.17).

Remarks:

1. The POF corresponding to this step will be $\beta_A = \beta$. It is set to be $\frac{\delta}{7}$, as there are in total seven β terms in the algorithm.
2. There is a possibility that a few scenarios might lead to infeasibility of the scenario Problem (4.17). In such circumstances, the optimisation schemes introduced in Chapter 3 (Section 3.5) can be used.

Step B (Test the solution u^* against the VaR constraint, and if required, check whether the solution can be improved)

1. Use the violation probability ϵ_V (from Problem (4.14)), the desired confidence parameter $\beta_V = \frac{\delta}{7}$ and the margin term ϱ in the procedure $\text{Find_N_Test}(\epsilon_V, \varrho, \beta_V)$, to determine the required number of scenarios N_T for the test ahead.
2. Generate N_T scenarios of $\mathbf{w} \in \mathbb{W}$, and compute the following violation probability estimate,

$$\hat{p} = \frac{1}{N_T} \sum_{k=1}^{N_T} \mathbb{1}\{\bar{g}(u^*, \mathbf{w}^{(k)}) > 0\}, \quad (4.31)$$

where $\mathbb{1}(\cdot)$ denotes the indicator function.

3. Check if $\hat{p} \leq \epsilon_V - \varrho$, i.e. whether the test is passed, if yes, then exit the algorithm

and forward the following solution,

$$\left\{ \begin{array}{l} \mathbf{u}^* \\ \text{with a certificate: } \mathbb{P}\{\mathbf{w} \in \mathbb{W} : \mathbf{u}^* \in \mathcal{U} \cap \bar{f}(\mathbf{u}^*, \mathbf{w}) \leq 0\} \geq 1 - \epsilon, \\ \mathbb{P}\{\mathbf{w}_U \in \mathbb{W} : \bar{g}(\mathbf{u}^*, \mathbf{w}) \leq 0\} \geq 1 - \epsilon_V. \end{array} \right.$$

Otherwise, test the validity of the *default solution* $\mathbf{u}_0^* \in \mathcal{U}'$ (see Section 4.2.1) against the constraint ' $\bar{g}(\mathbf{u}_0^*, \mathbf{w}^{(k)}) \leq 0$ ', for $k = 1, 2, \dots, N_T$, by computing the following violation probability estimate,

$$\hat{p}_0 = \frac{1}{N_T} \sum_{k=1}^{N_T} \mathbb{1}\{\bar{g}(\mathbf{u}_0^*, \mathbf{w}^{(k)}) > 0\}. \quad (4.32)$$

If the test passes, i.e. $\hat{p}_0 \leq \epsilon_V - \varrho$, then proceed to the next step. Otherwise, exit the algorithm and inform the system operators.

4. Save, in a set \mathcal{Q} , the \mathbf{w} -scenarios which violate the constraint ' $\bar{g}(\mathbf{u}, \mathbf{w}) \leq 0$ ' with \mathbf{u}^* , but satisfy the constraint with \mathbf{u}_0^* , i.e. $\mathcal{Q} = \{\mathbf{w}^{(k)} | \bar{g}(\mathbf{u}^*, \mathbf{w}^{(k)}) > 0 \wedge \bar{g}(\mathbf{u}_0^*, \mathbf{w}^{(k)}) \leq 0\}$ for $k = 1, 2, \dots, N_T$.

5. Compute the following probability estimate,

$$\hat{p}_T = \frac{|\mathcal{Q}|}{N_T}, \quad (4.33)$$

where $|\mathcal{Q}|$ is the cardinality of the set \mathcal{Q} .

6. Calculate the upper bounds $\hat{\epsilon}_T$ and $\hat{\epsilon}_0$ on the violation probabilities p_T and p_0 using the estimates \hat{p}_T and \hat{p}_0 , i.e. $\hat{\epsilon}_T = \text{Upper_Bound}(\hat{p}_T, \delta/7, N_T)$ and $\hat{\epsilon}_0 = \text{Upper_Bound}(\hat{p}_0, \delta/7, N_T)$.

Remarks:

1. The POF of this step corresponding to the two tests and two upper bounds will be $\beta_B = \frac{4}{7}\delta$.

4.3 Optimisation and Testing algorithm to solve an M-CCP with 2 chance-constraints 175

2. The set \mathcal{Q} , in Step B-4, will be used in the next step. It is a subset of the set $\mathcal{T} = \{\mathbf{w} \in \mathbb{W} \mid \bar{g}(\mathbf{u}^*, \mathbf{w}) > 0 \wedge \bar{g}(\mathbf{u}_0^*, \mathbf{w}) \leq 0\}$ (for details, see Eq. 4.27, and its explanation).
3. The probability estimate \hat{p}_T , in Step B-5, is the fraction of the N_T drawn \mathbf{w} -samples, which violates the constraint, ' $\bar{g}(\mathbf{u}, \mathbf{w}^{(k)}) \leq 0$ ' with $\mathbf{u} = \mathbf{u}^*$, but satisfy with $\mathbf{u} = \mathbf{u}_0^*$.
4. The upper bounds in Step B-6 ensure that $\mathbb{P}(p_T < \hat{\epsilon}_T) \geq 1 - \delta/7$ and $\mathbb{P}(p_0 < \hat{\epsilon}_0) \geq 1 - \delta/7$, and they will also be used in the next step.
5. If the algorithm exits in Step B-3, then the control can be shifted to an emergency operation depending on the given application.

Step C (Improve the solution \mathbf{u}^* in the direction of \mathbf{u}_0^* [125])

1. Use Eq. 4.34 to calculate the allowed (conditional) violation probability, ϵ_α^3 , for the chance-constrained problem corresponding to the one-dimensional scenario Problem (4.26), that improves the solution \mathbf{u}^* in the direction of \mathbf{u}_0^* .

$$\epsilon_\alpha = \frac{\epsilon_V - \hat{\epsilon}_0}{\hat{\epsilon}_T}. \quad (4.34)$$

(Eq. 4.34 is obtained from Theorem 4.2, stated in Section 4.3.1).

2. Use the violation probability ϵ_α and the desired confidence parameter $\beta_\alpha = \frac{\delta}{7}$ in the procedure `Find_N_CC`($\epsilon_\alpha, \beta_\alpha, 1$) to determine the required number (N_α) of scenarios, $\mathbf{w} \in \mathcal{Q}$, for the scenario Problem (4.26).
3. If $N_\alpha > |\mathcal{Q}|$, then generate $N_\alpha - |\mathcal{Q}|$ realisations of $\mathbf{w} \in \mathbb{W}$, such that the constraint $\bar{g}(\mathbf{u}^*, \mathbf{w}) \leq 0$ is violated, but $\bar{g}(\mathbf{u}_0^*, \mathbf{w}) \leq 0$ is satisfied. Include the newly generated realisations of \mathbf{w} in \mathcal{Q} .
4. Define a variable $\alpha \in (0, 1]$ (the optimisation variable of Problem (4.26)), and consider a straight line that connects the two solutions: \mathbf{u}^* and \mathbf{u}_0^* , i.e. $\bar{\mathbf{u}} =$

³ ϵ_α is the allowed violation probability, conditioned on a set inside \mathbb{W} , where each \mathbf{w} violates $\bar{g}(\mathbf{u}^*, \mathbf{w}) \leq 0$, but satisfies $\bar{g}(\mathbf{u}_0^*, \mathbf{w}) \leq 0$.

$(1 - \alpha)\mathbf{u}^* + \alpha\mathbf{u}_0^*$. Solve Problem (4.26) by replacing \mathcal{T} with \mathcal{Q} , and save the solution in α^* .

5. Calculate the improved solution as $\bar{\mathbf{u}}^* = (1 - \alpha^*)\mathbf{u}^* + \alpha^*\mathbf{u}_0^*$.

(Next, check the validity of the solution $\bar{\mathbf{u}}^*$ against the first chance-constraint of Problem (4.14), as it was not included in the scenario Problem (4.26) while *improving*, and provide the final certificate on the solution).

6. Use the violation probability ϵ (from Problem (4.14)), the desired confidence parameter $\beta = \frac{\delta}{7}$ and the margin term ϱ in the procedure Find_N.Test(ϵ, ϱ, β), to determine the required number of scenarios N_I for the test ahead.

7. Generate N_I scenarios of $\mathbf{w} \in \mathbb{W}$, and compute the following violation probability estimate,

$$\hat{p}_I = \frac{1}{N_I} \sum_{k=1}^{N_I} \mathbb{1}\{\bar{f}(\bar{\mathbf{u}}^*, \mathbf{w}^{(k)}) > 0\}. \quad (4.35)$$

8. Check if $\hat{p}_I \leq \epsilon - \varrho$, i.e. whether the test is passed, if yes, then exit the algorithm and forward the following solution,

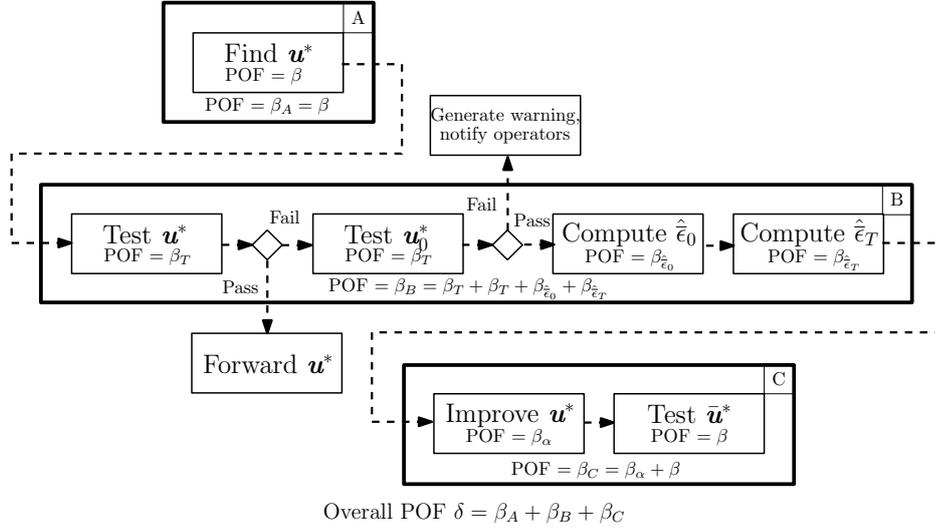
$$\left\{ \begin{array}{l} \bar{\mathbf{u}}^* \\ \text{with a certificate: } \quad \mathbb{P}\{\mathbf{w} \in \mathbb{W} : \bar{\mathbf{u}}^* \in \mathcal{U}' \cap \bar{f}(\bar{\mathbf{u}}^*, \mathbf{w}) \leq 0\} \geq 1 - \epsilon, \\ \quad \quad \quad \mathbb{P}\{\mathbf{w} \in \mathbb{W} : \bar{g}(\bar{\mathbf{u}}^*, \mathbf{w}) \leq 0\} \geq 1 - \epsilon_V, \end{array} \right.$$

otherwise, exit the algorithm with the following solution,

$$\left\{ \begin{array}{l} \bar{\mathbf{u}}^* \\ \text{with a certificate: } \quad \mathbb{P}\{\mathbf{w} \in \mathbb{W} : \bar{\mathbf{u}}^* \in \mathcal{U}' \cap \bar{g}(\bar{\mathbf{u}}^*, \mathbf{w}) \leq 0\} \geq 1 - \epsilon_V. \end{array} \right.$$

Remarks:

1. The POF of this step corresponding to the scenario problem and the test will be $\beta_C = \frac{2}{7}\delta$.


 Figure 4.6: *Optimisation and Testing algorithm.*

2. We use Eq. 4.34 to find the allowed violation probability ϵ_α for the chance-constrained problem, corresponding to the one-dimensional scenario Problem (4.26). In a rare occasion, it might happen that ϵ_α comes out to be very small, which will correspond to a very large number of scenarios N_α , based on Eq. 3.24. If N_α is more than the maximum number of computationally allowed constraints in the one dimensional Problem (4.26), say N_{\max} , then we replace N_α with N_{\max} . However, in such circumstances, the final solution is tested against the second (risk mitigation) chance-constraint of Problem (4.14). If the solution passes the test, then the probabilistic guarantees of the solution of Problem (4.26) will be valid, otherwise, we will lose the guarantees. It is a rare case as Problem (4.26) is a one-dimensional problem and N_{\max} is expected to be large. However, N_{\max} and the computational power required to solve the one-dimensional Problem (4.26) is different for different systems. It also depends on factors such as the number of states in the state-space description of the system (Eq. 4.1), horizon length M etc.

Fig. 4.6 summarises the algorithm steps and their POFs. The overall POF of the algorithm, as indicated in Fig. 4.6, will be the sum of the POFs of each step, i.e. $\delta = \beta_A + \beta_B + \beta_C = 7\beta$.

General remarks and benefits

The following remarks can be made on the proposed algorithm to find an approximate solution to M-CCPs,

1. The chance-constraints in Problem (4.14) can require strong probabilistic bounds, e.g. ϵ_V can be very small, in the order of 10^{-3} . It will then require a large number of scenarios to solve the problem by the scenario approach which can be computationally expensive. To compensate for that, this algorithm performs a relatively faster optimisation (Problem (4.17), i.e. without the second chance-constraint in Problem (4.14)), does some computationally cheap tests and solves a one dimensional optimisation problem (Problem (4.26)). These steps together are expected to be faster than the time to solve Problem (4.14) using other techniques. This is because the other techniques apply the scenario approach to the second chance-constraint as well (with the small probabilistic bound ϵ_V), and the corresponding scenario optimisation problem will then require a large number of constraints and demand high computational power.
2. ϵ and ϵ_V are the allowed violation probabilities over the time horizon M , used in the S-MPC based Problem (4.14). In practice, the selection is not easy, because operators are usually more interested and experienced to provide the violation probabilities over a much longer period, e.g. a year. Moreover, due to the dependence between the S-MPC problems solved at each time step, there is no easy way to infer ϵ and ϵ_V from the available information on the violation probability bounds for a longer period. Thus the selection of these probabilities, in practice, is more of a guess work, and it improves with experiments.
3. A sampling-and-discarding approach ([29]) can be easily introduced in the proposed algorithm to deal with possible conservatism in ensuring feasibility of the scenario problem constraints, at the price of cost function. In this approach, the constraints of the problem can be removed, to further minimise the cost function, with a decrease in feasibility up to an allowed limit, which can be obtained from the bounds ϵ and ϵ_V .

4.4 M-CCPs with more than two chance-constraints - an extension

The algorithm in Section 4.3.2 is developed for M-CCPs with two (2) chance-constraints. However, the idea of *optimisation*, *testing* and *improving*, can be extended to an M-CCP with any finite number of chance-constraints, and it is discussed in this section. The idea of the extension and the associated algorithm are built upon the results and procedures in Section 4.3.

An M-CCP problem with n_C chance-constraints is given by,

$$\begin{aligned} & \min_{\Gamma, \Theta} J(\mathbf{u}, \mathbf{w}), & (4.36) \\ \text{s.t.} \quad & \mathbb{P}\{\mathbf{w} \in \mathbb{W} : \mathbf{u}(\mathbf{w}) \in \mathcal{U} \cap \bar{f}_1(\mathbf{u}, \mathbf{w}) \leq 0\} \geq 1 - \epsilon_1, \\ & \mathbb{P}\{\mathbf{w} \in \mathbb{W} : \bar{f}_i(\mathbf{u}, \mathbf{w}) \leq 0\} \geq 1 - \epsilon_i, \quad \text{for } i = 2, 3, \dots, n_C, \end{aligned}$$

subject to the states dynamics, \mathbf{x} in Eq. 4.9, and the parametrisation of control policies, \mathbf{u} in Eq. 4.15. The notations are the same as in Problem (4.14), and we assume $\bar{f}_i(\mathbf{u}, \mathbf{w})$ (where $i = 1, 2, \dots, n_C$) are convex functions with respect to the optimisation variables (Γ and Θ), that define the control policy $\mathbf{u}(\mathbf{w})$, see Eq. 4.15. Also, we assume the states dynamics are incorporated in the objective function $J(\mathbf{u}, \mathbf{w})$ (e.g. see Eqs. 3.52 & 3.53). Furthermore, we assume $\epsilon_{n_C} < \epsilon_{n_C-1} < \dots < \epsilon_1$.

The intuition behind the extension is as follows: we first solve the scenario version of Problem (4.36) with the first chance-constraint only⁴ (i.e. the constraint on $\mathbf{u}(\mathbf{w})$ and $\bar{f}_1(\mathbf{u}, \mathbf{w})$ only), and then test the obtained solution, \mathbf{u}_1^* , against the other $n_C - 1$ chance-constraints. The first chance-constraint is used in the scenario problem, because ϵ_1 is the largest allowed violation probability among the ϵ_i s, where $i = 1, 2, \dots, n_C$. ϵ_1 will require the least number of constraints in the scenario problem. The number of scenarios can be found using the procedure `Find_N_CC`(ϵ_1, β, d) in Section 4.3.1. For *testing*, we use the procedure `Find_N_Test`($\epsilon_i, \varrho, \beta$) in Section 4.3.1 (where $i = 2, 3, \dots, n_C$ and $\varrho \in (0, \epsilon_{n_C})$ is a safety margin), to calculate the number of realisations (N_{T_i}) of \mathbf{w} , required to test the

⁴There is a possibility that a few scenarios might lead to infeasibility of the scenario problem. In such circumstances, the optimisation schemes introduced in Chapter 3 (Section 3.5) can be used.

i^{th} chance-constraint, and then calculate the empirical violation probabilities \hat{p}_i for each chance-constraint. There can be 2^{n_C-1} possible outcomes of the tests,

$$\begin{aligned} &P_1P_2P_3 \cdots P_{n_C} \\ &P_1P_2P_3 \cdots F_{n_C} \\ &\quad \vdots \\ &P_1F_2F_3 \cdots P_{n_C} \\ &P_1F_2F_3 \cdots F_{n_C} \end{aligned}$$

where P_i or F_i , for $i = 1, 2, \dots, n_C$, are the components of the test outcome. P_i indicates that the i^{th} chance-constraint has passed the test and F_i indicates that it has failed the test. The first component of the outcome (i.e. P_1) is assumed to be always P at this stage, because we solved the scenario optimisation problem prior to *testing*, to make sure that the first chance-constraint is satisfied.

We prioritise the chance-constraints (CC) in the following order,

$$CC_{n_C} > CC_{n_C-1} > \cdots > CC_2 > CC_1, \quad (4.37)$$

i.e. the priority of satisfying the n_C^{th} chance-constraint in Problem (4.36) is the highest⁵. Let us denote the test output, corresponding to the solution \mathbf{u}_1^* , as $P_1^1X_2^1X_3^1 \cdots X_{n_C}^1$, where X_i^1 can be P_i^1 or F_i^1 , and the superscript '1' will be used in the *improving* phase below. If we have $X_i^1 = P_i^1, \forall i = 2, 3, \dots, n_C$, then we exit the algorithm, and return the solution with a certificate guaranteeing satisfaction of all chance-constraints, otherwise, we improve the solution.

For *improving*, we first test the *default solution* $\mathbf{u}_0^* \in \mathcal{U}'$ (where $\mathcal{U} \subseteq \mathcal{U}'$, for details see Section 4.2.1) against the last $n_C - 1$ chance-constraints of Problem (4.36) in the same way as we tested \mathbf{u}_1^* above. Let us denote the test output as $P_1D_2D_3 \cdots D_{n_C}$, where again, D_i can be P_i or F_i based on whether the i^{th} chance-constraint passed or failed the test with \mathbf{u}_0^* . If $D_i = F_i, \forall i = 2, 3, \dots, n_C$, it means that the *default solution* failed the tests against

⁵In the following algorithm, we can use any user-chosen priority, but the components of the test outcome are required to be written in the order of increasing priority.

every chance-constraint in the problem, and there is no room for improvement. We exit the algorithm and some emergency or exceptional operations are followed. Otherwise, we will try to improve the chance-constraints, which failed the test with \mathbf{u}_1^* , but passed the test with \mathbf{u}_0^* , starting with the highest numbered chance-constraint (CC), i.e. based on their priority (Eq. 4.37). If we need to improve, e.g. ℓ^{th} chance-constraint, then we use Step-C of the *Optimisation and Testing algorithm* for *improving*, and in Eq. 4.34, we replace ϵ_V with ϵ_ℓ . We calculate the required number of scenarios N_α (using the procedure `Find_N_CC($\epsilon_\alpha, \beta, 1$)`), for the one-dimensional (*improving*) scenario problem, similar to Problem (4.26), with the ℓ^{th} chance-constraint of Problem (4.36). After the improvement is made, we test the improved solution, \mathbf{u}_2^* , against all the other chance-constraints in Problem (4.36), in the same way as we tested \mathbf{u}_1^* before. Let us denote the test output as $X_1^2 X_2^2 X_3^2 \cdots X_{n_C}^2$, where again, X_i^2 can be P_i^2 or F_i^2 , based on whether the i^{th} chance-constraint in Problem (4.36) passed or failed the test with \mathbf{u}_2^* .

The solution can be further analysed and improved in an iterative way. Considering k as the iteration index, whereas in the last improvement k was 1. There are the following three possibilities related to the test outcomes on the improved solutions, with their corresponding actions, in these iterations,

1. If we achieve $X_i^{k+1} = P_i^{k+1}, \forall i = 2, 3, \dots, n_C$ and $k \geq 1$, then we exit the algorithm, and return the solution \mathbf{u}_k^* , with a certificate guaranteeing satisfaction of all chance-constraints.
2. If we achieve a test outcome where any higher numbered chance-constraint's component changes from P to F , we exit the algorithm, and return the solution obtained in the previous step, i.e. \mathbf{u}_k^* . This is done to preserve the priorities of the chance-constraints (CC), as in Eq. 4.37. The solution is returned with a certificate guaranteeing satisfaction of the chance-constraints, whose related components in the corresponding test outcome, i.e. $X_1^k X_2^k X_3^k \cdots X_{n_C}^k$, were P .
3. If we achieve a test outcome where no higher numbered chance-constraint's component changes from P to F , then we consider the next highest numbered chance-constraint (compared to the one which we improved in the previous iteration) for

improvement, among the chance-constraints which were identified for possible improvement in the test outcome, i.e. they failed the test with \mathbf{u}_k^* , but passed the test with \mathbf{u}_0^* . The improvement is then followed by *re-testing*, with an incremented k value in the test outcome. However, if there is no such chance-constraint left for improvement, then we exit the algorithm and return the solution \mathbf{u}_k^* , with a certificated guaranteeing satisfaction of the chance-constraints, whose related components in the corresponding test outcome, i.e. $X_1^k X_2^k X_3^k \dots X_{n_C}^k$, were P . If the algorithm does not exit, then the iterations continue and the algorithm can have, by design, at most $n_C - 1$ iterations.

The confidence parameter β , corresponding to a scenario problem, test, or the calculation of an upper bound (Section 4.3.1) in the algorithm can be calculated as, $\beta = \delta / ((n_C - 1)^2 + 5(n_C - 1) + 1)$, where δ is the overall Probability Of Failure (POF) of the algorithm (see Section 4.3.1). The expression ' $\delta = ((n_C - 1)^2 + 5(n_C - 1) + 1)\beta$ ' is explained next. The algorithm consists of the the following number of β terms in the order of their occurrences,

- 1 term for the first scenario problem, i.e. the scenario problem corresponding to Problem (4.36) with the first chance-constraint only.
- $2(n_C - 1)$ terms for $2(n_C - 1)$ tests on the remaining $n_C - 1$ constraints of Problem (4.36) with \mathbf{u}_1^* and \mathbf{u}_0^* respectively.
- $(n_C - 1)(2 + 1 + (n_C - 1))$ terms for at most $n_C - 1$ improvements, where in each improvement 2 upper bounds are computed, 1 one-dimensional scenario problem is solved and $n_C - 1$ tests are performed against $\bar{\mathbf{u}}_{k+1}^*$, where k is the index of improvement, and $1 \leq k < n_C$.

The total number of β terms in the algorithm are therefore equal to $((n_C - 1)^2 + 5(n_C - 1) + 1)$. Next, we present the algorithm formally.

Algorithm

1. Select the overall POF, δ , to be a very small value (e.g. 10^{-8}), and define the confidence parameter, $\beta = \delta / ((n_C - 1)^2 + 5(n_C - 1) + 1)$.
2. Compute $N_r = \text{Find_N_CC}(\epsilon_1, \beta, d)$, where ϵ_1 is the allowed violation of the first chance-constraint in Problem (4.36) and d is the number of optimisation variables in the problem.
3. Solve the scenario version of Problem (4.36) with the first chance-constraint only, i.e. consider the constraints: $\mathbf{u}(\mathbf{w}^{(j)}) \in \mathcal{U}$ and $\bar{f}_1(\mathbf{u}(\mathbf{w}^{(j)}), \mathbf{w}^{(j)}) \leq 0$, for $j = 1, 2, \dots, N_r$, in the scenario problem. Save the solution in \mathbf{u}_1^* .
4. Compute $N_{T_i} = \text{Find_N_Test}(\epsilon_i, \varrho, \beta)$, where ϵ_i , for $i = 2, 3, \dots, n_C$, are the allowed violation probabilities of the last $n_C - 1$ chance-constraints in Problem (4.36), and $\varrho \in (0, \epsilon_{n_C})$ is a safety margin (e.g. $\varrho = \epsilon_{n_C}/2$).
5. Generate $N_{T_{n_C}}$ realisations of $\mathbf{w} \in \mathbb{W}$ for *testing* purposes.
6. Compute $\hat{p}_i = \frac{1}{N_{T_i}} \sum_{k=1}^{N_{T_i}} \mathbb{1}\{\bar{f}_i(\mathbf{u}_1^*, \mathbf{w}^{(k)}) > 0\}$, for $i = 2, 3, \dots, n_C$.
7. If $\hat{p}_i \leq \epsilon_i - \varrho_i$, assign $X_i^1 = P_i^1$ in the test outcome, otherwise, assign $X_i^1 = F_i^1$. Save the test outcome as $P_1^1 X_2^1 X_3^1 \cdots X_{n_C}^1$.
8. If $X_i^1 = P_i^1, \forall i = 2, 3, \dots, n_C$, then exit the algorithm and return the solution \mathbf{u}_1^* , with a certificate guaranteeing satisfaction of all chance-constraints in Problem (4.36).
9. Otherwise, if there is an F component in the test outcome, test the *default solution*, \mathbf{u}_0^* against the last $n_C - 1$ constraints in Problem (4.36), the same way as \mathbf{u}_1^* was tested above in Step 7. Save the computed violation probability estimates as \hat{p}_{0_i} , for $i = 2, 3, \dots, n_C$, and the test outcome as $P_1 D_2 D_3 \cdots D_{n_C}$.
10. If $D_i = F_i, \forall i = 2, 3, \dots, n_C$, then exit the algorithm and alert the operators to initiate emergency or exceptional operations, because the *default solution* failed all the chance-constraints and therefore there is no possible room of improvement on \mathbf{u}_1^* .

11. Otherwise, if there are P_i components, with $i > 1$, in the test outcome, $P_1 D_2 D_3 \cdots D_{n_C}$, improvement can be made on the solution \mathbf{u}_1^* . Initialise ℓ to be the index of the first F component in the test outcome, $P_1^1 X_2^1 X_3^1 \cdots X_{n_C}^1$, from right to left, provided the corresponding ℓ^{th} component in the test outcome, $P_1 D_2 D_3 \cdots D_{n_C}$, is a P , i.e.

$$\ell = \max\{i | X_i^1 = F \wedge D_i = P\},$$

where $i = 2, 3, \dots, n_C$.

12. Initialise $k = 1$ as the index of improvement, and run the following steps in a while loop, with the true condition: $\ell > 1$.

- (a) Compute ϵ_α using Eq. 4.34, with $\epsilon_V = \epsilon_\ell$, $\hat{\epsilon}_T = \hat{\epsilon}_{T_\ell} = \text{Upper_Bound}(\hat{p}_\ell, \beta, N_{T_\ell})$, and $\hat{\epsilon}_0 = \hat{\epsilon}_{0_\ell} = \text{Upper_Bound}(\hat{p}_{0_\ell}, \beta, N_{T_\ell})$.
- (b) Compute $N_\alpha = \text{Find_N_CC}(\epsilon_\alpha, \beta, 1)$.
- (c) Improve \mathbf{u}_k^* in the direction of \mathbf{u}_0^* by solving the following one dimensional (*improving*) scenario problem (similar to Problem (4.26)),

$$\begin{aligned} & \min_{\alpha} J(\bar{\mathbf{u}}), & (4.38) \\ \text{s.t. } & \mathbf{u}(\mathbf{w}^{(j)}) \in \mathcal{U}', \quad \bar{f}_\ell(\mathbf{u}, \mathbf{w}^{(j)}) \leq 0, \\ & \text{for } j = 1, 2, \dots, N_\alpha, \quad \mathbf{w}^{(j)} \in \mathcal{Q}, \end{aligned}$$

where $\mathbf{u} = (1 - \alpha)\mathbf{u}_k^* + \alpha\mathbf{u}_0^*$ and $\mathcal{Q} = \{\mathbf{w}^{(j)} | \bar{f}_\ell(\mathbf{u}_k^*, \mathbf{w}^{(j)}) > 0 \wedge \bar{f}_\ell(\mathbf{u}_0^*, \mathbf{w}^{(j)}) \leq 0\}$, for $j = 1, 2, \dots, N_{T_\ell}$. If $|\mathcal{Q}| < N_\alpha$, where $|\mathcal{Q}|$ is the cardinality of the set \mathcal{Q} , then generate the remaining $N_\alpha - |\mathcal{Q}|$ realisations of $\mathbf{w} \in \mathbb{W}$ and put them in \mathcal{Q} ,

- (d) Use the problem output, α^* , to define the improved solution, $\mathbf{u}_{k+1} = (1 - \alpha^*)\mathbf{u}_k^* + \alpha^*\mathbf{u}_0^*$.
- (e) Test the solution \mathbf{u}_{k+1}^* against the i^{th} chance-constraints in Problem (4.36), where $i = \{1, 2, \dots, n_C\} \setminus \ell$, the same way as \mathbf{u}_1^* was tested above in Step 7. Generate the test outcome as $X_1^{k+1} X_2^{k+1} X_3^{k+1} \cdots X_{n_C}^{k+1}$.

- (f) If $X_i^{k+1} = P_i^{k+1}, \forall i = \{1, 2, \dots, n_C\} \setminus \ell$, then exit the algorithm and return the solution \mathbf{u}_{k+1} , with a certificate guaranteeing satisfaction of all chance-constraints in Problem (4.36).
- (g) If any P_i^k component changes to F_i^{k+1} , for $i > \ell$, then exit the algorithm and return the solution \mathbf{u}_k^* , obtained prior to the improvement, with a certificate guaranteeing satisfaction of those chance-constraints in Problem (4.36), whose components in the test outcome, $X_1^k X_2^k X_3^k \cdots X_{n_C}^k$, were P . (This is done to preserve priorities of the chance-constraints (CC), as in Eq. 4.37).
- (h) Otherwise, rewrite ℓ to be the index of the first F component in the test outcome, $X_1^{k+1} X_2^{k+1} X_3^{k+1} \cdots X_{n_C}^{k+1}$, to the left of X_ℓ^{k+1} , provided the corresponding ℓ^{th} component in the test outcome, $P_1 D_2 D_3 \cdots D_{n_C}$ is a P , i.e.

$$\ell = \max\{i | X_i^{k+1} = F \wedge D_i = P\},$$

where $i = 2, 3, \dots, \ell - 1$.

- (i) Increment k ($k = k + 1$), and check the while loop condition, i.e. if $\ell > 1$, jump to Step 12-a, otherwise, exit the loop.

13. Return the solution \mathbf{u}_k^* , with with a certificate guaranteeing satisfaction of those chance-constraints in Problem (4.36), whose components in the test come $X_1^k X_2^k X_3^k \cdots X_{n_C}^k$ are P .

The algorithm solves M-CCPs with a finite number of chance-constraints, using the results and procedures in Section 4.3. The algorithm can also be used in an MPC setting, and it provides probabilistic guarantees on the solutions. All optimisation problems in the algorithm are assumed to be feasible, however, if required and as discussed earlier, the feasibility assurance schemes introduced in Chapter 3 can be used. All (*improving*) scenario problems in the algorithm, which are at max $n_C - 1$, are one dimensional problems. Therefore, they are also expected to keep the computational burden of the algorithm reasonable.

4.5 A simulation based example

In this section we address a simple control problem with two chance-constraints. We consider a linear system with one output y (units m) and three inputs u_A , u_B and u_C (units m^3/sec). The inputs and the output are related by the following discrete time (data-based) model,

$$y(n+1) = y(n) + 5.57 \times 10^{-3}[u_A(n-2) + u_B(n-1) - u_C(n)], \quad (4.39)$$

where the input u_A is regulated and it acts with a delay of two time steps (one time step $T_s = 1,200$ sec), the input u_B is unregulated, forecast-able and measurable, it acts with a unit step delay, and u_C is a constant input. Our aim is to control the output, y , between an upper and a lower limit in the presence of a large unregulated input u_B . We formulate the control problem as an M-CCP with two chance-constraints, based on the control objectives (stated below), and use the *Optimisation and Testing algorithm* (Section 4.3.2) to solve the M-CCP within a Stochastic MPC setting.

4.5.1 Control objectives and control design

Here we first describe the two chance-constraints and the objective function related to the control problem, which form an M-CCP, and then parametrise the *Optimisation and Testing algorithm* to solve the M-CCP.

During normal operations, the following constraints are required to be satisfied with probability at least $1 - \epsilon$, where ϵ is 0.1,

- $20 \leq u_A(n) \leq 70$, i.e. the input u_A can vary between 20–70 m^3/sec ,
- $-5 \leq u_A(n) - u_A(n-1) \leq 5$, i.e. the change in the input u_A should be less than 5 m^3/sec ,
- $4.75 \leq y(n) \leq 5.65$, i.e. the output should stay between 4.75–5.65 m . Ideally, it should stay at the mean value, i.e. 5.2 m .

Similarly, we consider the following constraint to be satisfied with a higher probability, i.e. at least $1 - \epsilon_V$, where ϵ_V is 0.001. The constraint is a risk mitigation constraint,

- $y(n) \leq 5.85$.

u_C is a fixed input of $85 \text{ m}^3/\text{sec}$, and we consider the *default solution*, $u_{A,0}^* = 10 \text{ m}^3/\text{sec}$, i.e. we are allowed to reduce u_A below the lower limit of $20 \text{ m}^3/\text{sec}$ in order to avoid the output, y , to cross a higher risk related limit of 5.85 m .

In a Stochastic MPC setting, we solve an optimisation problem over a finite time horizon in the future. Therefore, in the current control problem, we would need a forecast of the unregulated input u_B . In the following simulations we considered the forecast, u_B^f , as the lowpass filtered version of the actual input u_B , and relate them as

$$u_B(n) = u_B^f(n) + w_B(n),$$

where we used the filter as a first order, zero-phase Butterworth filter with cut off frequency $0.03 (T_s)^{-1}$. The disturbance term w_B is generated from a first order AR process $w_B(n+1) = aw_B(n) + e(n)$, with $a = 0.9$, and $\{e(n)\}$ is a sequence of independent and identically distributed zero mean Gaussian noises with standard deviation $\sigma = 2 \text{ m}^3/\text{sec}$.

In the M-CCP we use the following objective function (similar to Eq. 3.52 with $\mathbf{S} = \mathbf{0}$),

$$J(\mathbf{u}_A(\mathbf{w}_B)) = \mathbb{E}\left[\sum_{n=i}^{i+M-1} ((y(n) - y_r)^\top \mathbf{Q}(y(n) - y_r) + u_A(n)^\top \mathbf{R}u_A(n))\right], \quad (4.40)$$

where $M = 10$ is the horizon length, $y_r = 5.2$ is the reference output value. \mathbf{w}_B is a vector of w_B values over the finite horizon M , i.e. $\mathbf{w}_B = \begin{bmatrix} w_B(n) & w_B(n+1) & \dots & w_B(n+M-1) \end{bmatrix}^\top$. The notations are followed from Section 3.2.1, and from now on, all bold variables are vectors of the same variable defined over the finite horizon, M , like \mathbf{w}_B defined above. The notation $\mathbf{u}_A(\mathbf{w}_B)$ shows that the objective function is minimised over a set of control policies parametrised in \mathbf{w}_B . We use the affine parametrisation of \mathbf{u}_A in \mathbf{w}_B , as in Eq. 4.15, i.e. $\mathbf{u}_A = \Theta \mathbf{w}_B + \Gamma$, where Θ and Γ matrices form the optimisation variables. We selected \mathbf{Q} and \mathbf{R} matrices in Eq. 4.40 as block diagonal matrices with $Q = 10$ and $R = 5 \times 10^{-4}$ on the diagonals.

In the *Optimisation and Testing algorithm*, we selected the Overall POF, $\delta = 1 \times 10^{-6}$, and thus $\beta = \delta/7$. For $M = 10$, we got $d = 20$ optimisation variables while using the parametrisation of Θ and Γ matrices as in Eq. 3.19, for more details on how to calculate d , see the example in Section 3.6.1. The number of scenarios N_r for the scenario problem in the *optimisation* step (similar to Problem (4.16), where we ignore the risk related constraint) should be at least 730 (from Eq. 3.24). After finding the scenario solution, \mathbf{u}_A^* , we tested the risk related constraint, i.e. ' $\mathbf{y}(\mathbf{u}_A^*, \tilde{\mathbf{w}}_B^{(k)}) > 5.85$ ', where $k = 1, \dots, N_T$ and we used N_T new realisations of the \mathbf{w}_B variable. We selected $\rho = 0.0005$, and with $\epsilon_V = 0.001$, we got $N_T = 873,918$ (from Eq. 4.23). If the test passed, we forwarded the solution. Otherwise, we tested the *default solution* $u_{A,0}^* = 10 \text{ m}^3/\text{sec}$, and if the test passed (which always happened in the following simulations), we solved a one dimensional problem (similar to Problem (4.38)) to improve the solution \mathbf{u}_A^* in the direction of $\mathbf{u}_{A,0}^*$. In the one dimensional problem we minimised the same objective function $J(\bar{\mathbf{u}}_A)$ (Eq. 4.40, where $\bar{u}_A(n) = (1 - \alpha)u_A^*(n) + \alpha u_{A,0}^*$), with respect to α , subject to the following constraints,

- $10 \leq \bar{u}_A(n) \leq 80$, i.e. we relaxed the lower limit on the input u_A by $10 \text{ m}^3/\text{sec}$,
- $-5 \leq \bar{u}_A(n) - \bar{u}_A(n-1) \leq 5$,
- $4.75 \leq y(n) \leq 5.85$, where the upper limit on the output, y , is now 5.85.

The number of scenarios, N_α , for this problem was selected based on the relation in Eq. 4.29, for details see 'Background' in the third procedure in Section 4.3.1. All optimisation problems were solved by running YALMIP [3] over SDPT3 [4].

4.5.2 Simulation results

Here we compare the simulation results (Fig. 4.7) of the controlled output, with and without the use of the *Optimisation and Testing algorithm*. In the top graph of Fig. 4.7, the black straight line shows the desired output, $y_r = 5.2 \text{ m}$, and the blue and red dotted straight lines show the normal limits (4.75–5.65 m) and the higher risk related limit (5.85 m) on the output, y , respectively. The bottom graph shows the unregulated input, u_B , with a black curve, and we can see that due to the high input between the 70th and 120th

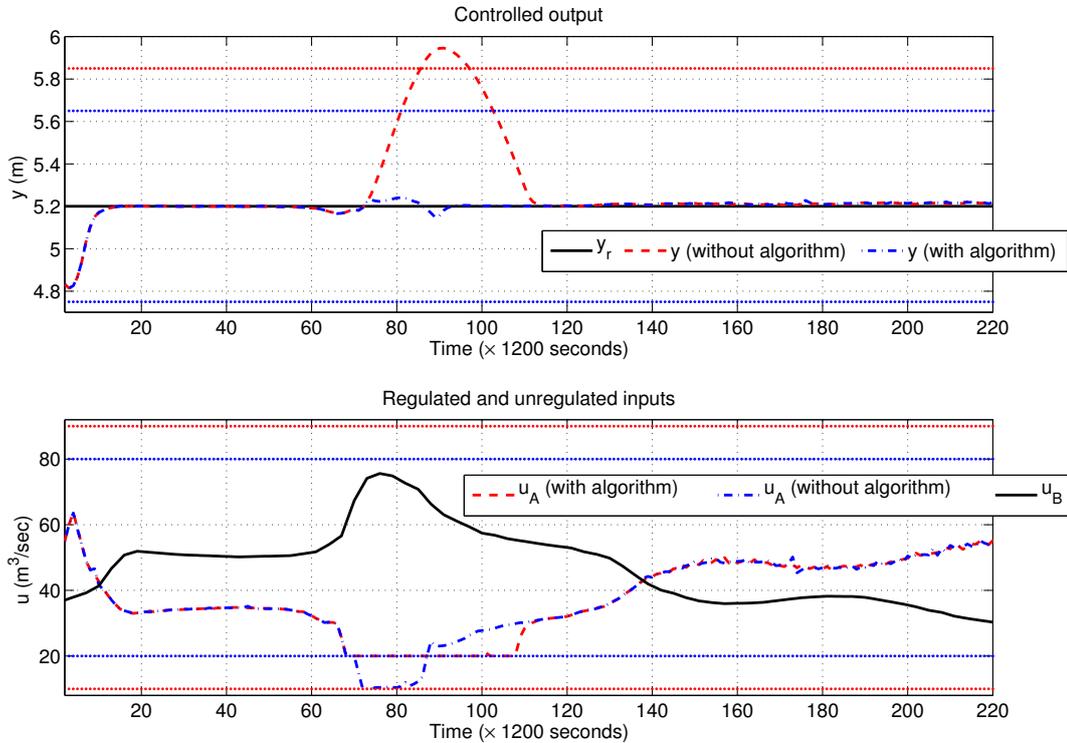


Figure 4.7: Control of the output y .

time steps, the controlled output (without the *Optimisation and Testing algorithm*, the red dashed curve in the top graph) shows an undesirable response. The output rises above the allowed limit, i.e. 5.65 m, and even above the risk related limit, i.e. 5.85 m. However, the blue dash-dotted curve shows the controlled output obtained from the algorithm, and the output stayed close to the reference value, i.e. 5.2 m, and is well within the limits throughout the simulations.

The bottom graph in the figure also shows the regulated input u_A . The blue and red dotted straight lines show the original and relaxed limits on u_A respectively. With the original constraints, in order to avoid the rise in the output y , due to the high unregulated input u_B , the input u_A was reduced to minimum. However, with the relaxed constraints, we had a choice to further reduce the input u_A by $10 \text{ m}^3/\text{sec}$, which was effective as shown in the bottom graph of Fig. 4.7 between 70^{th} and 90^{th} time steps, where the solution u_A^* failed the test against the risk related constraint, and the default solution $u_{A,0}^*$ was used

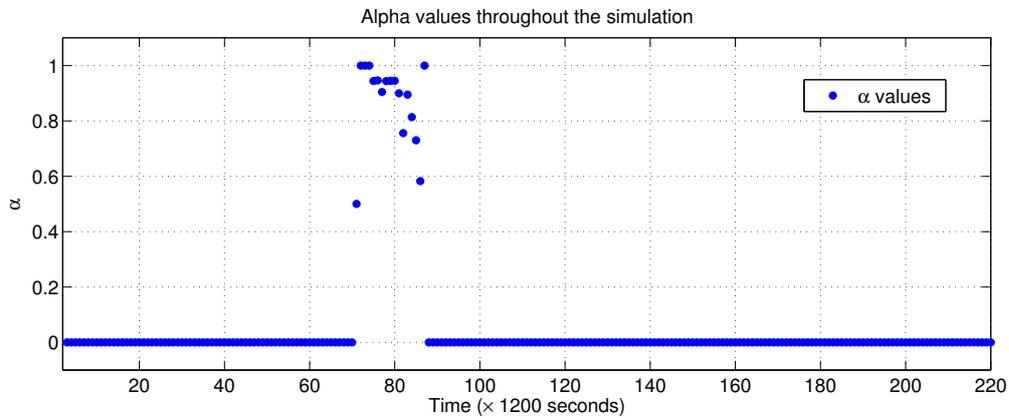


Figure 4.8: Alpha values computed by the algorithm (Section 4.3.2).

for improvement. Fig. 4.8 shows the values of α obtained during the simulation. There are non-zero values of α at the time steps where the improvements have been made. Moreover, the period where these improvements are made corresponds to the period when there are large u_B inputs, as shown in the bottom graph of Fig. 4.7.

Appendix

4.A Proof of Theorem 4.2

Fig. 4.9(a) shows the space \mathbb{W} from where we draw noise realisations \mathbf{w} . Let ‘ $\text{VS}(\mathbf{u}^*)$ ’ denotes the set that includes all realisations of $\mathbf{w} \in \mathbb{W}$ that violate the constraint ‘ $\bar{g}(\mathbf{u}^*, \mathbf{w}) \leq 0$ ’, where \mathbf{u}^* is a solution of Problem (4.17). In Fig. 4.9(a) the set $\text{VS}(\mathbf{u}^*)$ is shown as the blue area represented with dots. Similarly, we denote ‘ $\text{VS}(\mathbf{u}_0^*)$ ’ as the set that includes all realisations of $\mathbf{w} \in \mathbb{W}$ that violate the constraint ‘ $\bar{g}(\mathbf{u}_0^*, \mathbf{w}) \leq 0$ ’, where \mathbf{u}_0^* is the *default solution*. In Fig. 4.9(a) the set $\text{VS}(\mathbf{u}_0^*)$ is shown as the green area represented with crosses. Note that, $\text{VS}(\mathbf{u}_0^*)$ is not necessarily contained in $\text{VS}(\mathbf{u}^*)$. Fig. 4.9 shows both scenarios, in (a) the set $\{\text{VS}(\mathbf{u}_0^*) \setminus \text{VS}(\mathbf{u}^*)\}$ is non-empty, while in (b) the set $\text{VS}(\mathbf{u}_0^*)$ is contained in the set $\text{VS}(\mathbf{u}^*)$. The proof ahead is valid for both situations.

We define a set $\mathcal{T} = \{\text{VS}(\mathbf{u}^*) \setminus \text{VS}(\mathbf{u}_0^*)\}$ (Eq. 4.27) that contains all realisations of \mathbf{w} that strictly cause \mathbf{u}^* to violate the constraint ‘ $\bar{g}(\mathbf{u}^*, \mathbf{w}) \leq 0$ ’ and \mathbf{u}_0^* satisfies the corresponding constraint. In Problem (4.26), we specifically sample \mathbf{w} from the set \mathcal{T} and seek a solution α^* that improves the numerical value of the solution \mathbf{u}^* in the direction of \mathbf{u}_0^* . The set Q (in Step B-4 of the algorithm) is included in the set \mathcal{T} . The improved solution is obtained as $\bar{\mathbf{u}}^* = (1 - \alpha^*)\mathbf{u}^* + \alpha^*\mathbf{u}_0^*$. We define ‘ $\text{VS}(\bar{\mathbf{u}}^*)$ ’ as the set that includes all realisations of $\mathbf{w} \in \mathbb{W}$ that cause the improved solution ($\bar{\mathbf{u}}^*$) to violate the constraint ‘ $\bar{g}(\bar{\mathbf{u}}^*, \mathbf{w}) \leq 0$ ’. In Fig. 4.9 the set $\text{VS}(\bar{\mathbf{u}}^*)$ is shown with a red area represented with squares. By convexity, there is no realisation of \mathbf{w} for which the constraint ‘ $\bar{g}(\mathbf{u}, \mathbf{w}) \leq 0$ ’ is satisfied by both \mathbf{u}^* and \mathbf{u}_0^* and is violated by $\bar{\mathbf{u}}^*$. However, there can be realisations of \mathbf{w} that violate the aforementioned constraint with \mathbf{u}^* and/or \mathbf{u}_0^* but satisfy with $\bar{\mathbf{u}}^*$. The area representing

$VS(\bar{\mathbf{u}}^*)$ in Fig. 4.9 depicts such properties. Mathematically, we can represent it as

$$VS(\bar{\mathbf{u}}^*) = \{VS(\bar{\mathbf{u}}^*) \setminus \mathcal{T}\} \cup \{VS(\bar{\mathbf{u}}^*) \cap \mathcal{T}\}, \quad (4.41)$$

where $\mathcal{T} = \{VS(\mathbf{u}^*) \setminus VS(\mathbf{u}_0^*)\}$. Again, by the convexity argument stated above, if we subtract the set \mathcal{T} from $VS(\bar{\mathbf{u}}^*)$, the resulting set will be contained inside $VS(\mathbf{u}_0^*)$, because there is no realisation of \mathbf{w} for which the constraint ' $\bar{g}(\mathbf{u}, \mathbf{w}) \leq 0$ ' is satisfied by both \mathbf{u}^* and \mathbf{u}_0^* and is violated by $\bar{\mathbf{u}}^*$. Mathematically, we can represent it as $\{VS(\bar{\mathbf{u}}^*) \setminus \mathcal{T}\} \subseteq VS(\mathbf{u}_0^*)$, which is also supported by Fig. 4.9, and that gives us

$$VS(\bar{\mathbf{u}}^*) \subseteq VS(\mathbf{u}_0^*) \cup \{VS(\bar{\mathbf{u}}^*) \cap \mathcal{T}\}. \quad (4.42)$$

We can further write the probability of the set $VS(\bar{\mathbf{u}}^*)$ using Eq. 4.42 as

$$\mathbb{P}(VS(\bar{\mathbf{u}}^*)) \leq \mathbb{P}(VS(\mathbf{u}_0^*)) + \mathbb{P}(VS(\bar{\mathbf{u}}^*) \cap \mathcal{T}),$$

i.e.

$$\mathbb{P}(VS(\bar{\mathbf{u}}^*)) \leq \mathbb{P}(VS(\mathbf{u}_0^*)) + \mathbb{P}(VS(\bar{\mathbf{u}}^*)|\mathcal{T})\mathbb{P}(\mathcal{T}), \quad (4.43)$$

$$\mathbb{P}(VS(\bar{\mathbf{u}}^*)) \leq \hat{\epsilon}_0 + \mathbb{P}(VS(\bar{\mathbf{u}}^*)|\mathcal{T})\hat{\epsilon}_T, \quad (4.44)$$

considering $\mathbb{P}(VS(\mathbf{u}_0^*)) \leq \hat{\epsilon}_0$ and $\mathbb{P}(\mathcal{T}) \leq \hat{\epsilon}_T$. If we use the procedure `Find_N.Scenario`($\epsilon_\alpha, \beta_\alpha, 1$) to find the required number of scenarios N_α for Problem (4.26), then the scenario solution (α^*) of Problem (4.26), that gives the control input as $\bar{\mathbf{u}}^* = (1 - \alpha^*)\mathbf{u}^* + \alpha^*\mathbf{u}_0^*$, satisfies $\mathbb{P}(VS(\bar{\mathbf{u}}^*)|\mathcal{T}) \leq \epsilon_\alpha$ with a confidence $1 - \beta_\alpha$. Using

$$\epsilon_\alpha \leq \frac{\epsilon_V - \hat{\epsilon}_0}{\hat{\epsilon}_T}, \quad (4.45)$$

the theorem follows from Eq. 4.44.

□

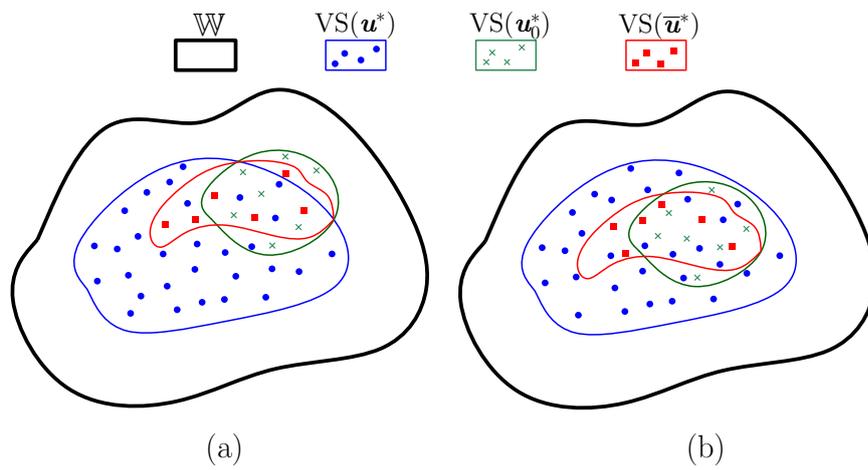


Figure 4.9: A pictorial description of violation sets.

This page intentionally left blank.

Chapter 5

Application of the Optimisation and Testing Algorithm to Flood Control

In Chapter 3 we presented control schemes for normal river operations, and the objectives were that water level in a reservoir and flow release from the reservoir should be kept within safe limits, while the change in flows and water levels should also be less than given thresholds. We also emphasised the fact that there are large distances between the points where the flows can be regulated in a river, and as a consequence, forecasts of the unregulated flows are required. Such forecasts are uncertain, and the control problem can be formulated as an optimisation problem with probabilistic constraints, such that a chance-constrained optimisation problem [24,26] is obtained (see Problem (3.55)). However, there is also a need to measure and mitigate flood risks, besides achieving normal river operations. Incorporation of flood risk mitigation in the existing river control problem is one of the aims of this chapter.

In Chapter 4 we formulated a Multiple Chance-Constrained optimisation Problem (M-CCP), where the control problem takes into account normal operations and the operations related to risk mitigation. We used Value-at-Risk (VaR) [86] as the risk measure. Constraints related to both operations were represented in two chance-constraints.

In this chapter we incorporate flood risk mitigation in the river control problem, formulated in Chapter 3, using the formulation of M-CCP with two chance-constraints, developed in Chapter 4. We use one chance-constraint to accommodate constraints related to normal river operations and the other chance-constraint corresponds to the VaR constraint which keeps the flood risks below a user-chosen threshold. We use the *Optimisation and Testing algorithm*, proposed in Section 4.3.2, to solve the M-CCP in an S-MPC

setting. The strategy of the algorithm, i.e. *optimisation*, *testing* and *improving*, suits flood risk mitigation problem well, because we do not want to be overly cautious about flood risks, since most of the time there is no or very little risk of flooding. In the algorithm, the chance-constraint associated with flooding is tested using the control actions/control policies obtained for normal operations. If the flood risk, detected by *testing*, is above the given threshold, we priorities flood risk mitigation over normal river operations, and modify the control actions/control policies.

We apply the algorithm to the historical operational data of the upper part of Murray River in Australia. We show that the algorithm not only achieves satisfactory normal river operations, but also mitigates flood risks from unregulated inflows, in a computationally cheap way.

The chapter is organised as follows. In Section 5.1 we consider a loss function for the flood control problem, and use it to develop a flood risk measure, which is later incorporated in the river control problem that tackles both normal river operations and flood risk mitigation. We then apply the developed formulation and the *Optimisation and Testing algorithm* to the operational data of the upper part of Murray River in Section 5.2.

5.1 Flood risk measure and mitigation

In this section we describe a loss function and a risk measure, which can be used for floods. We later use them to formulate a flood avoidance optimisation problem which can be solved in a Stochastic MPC setting.

5.1.1 Value-at-Risk as a flood risk measure

As described in Section 4.1.1, a loss function $g(u_C, w_U)$ is given in terms of decision variables $u_C \in U$ and uncertain variables $w_U \in W$. For rivers, $u_{C,n} = [Q_{cs,1}(n - \tau_{cs,1}), \dots, Q_{cs,n_{cs}}(n - \tau_{cs,n_{cs}}), Q_{cr,1}(n - \tau_{cr,1}), \dots, Q_{cr,n_{cr}}(n - \tau_{cr,n_{cr}})]^\top$, and $w_{U,n} = [Q_{us,1}(n - \tau_{us,1}), \dots, Q_{us,n_{cs}}(n - \tau_{us,n_{cs}}), Q_{ur,1}(n - \tau_{ur,1}), \dots, Q_{ur,n_{ur}}(n - \tau_{ur,n_{ur}})]^\top$, where $Q_{cs}(n)$ and $Q_{cr}(n)$ denote regulated in- and out-flows, and $Q_{us}(n)$ and $Q_{ur}(n)$ denote unregulated in- and out-flows. U and W are convex sets to which u_C and w_U be-

long, and τ represents the time delays corresponding to regulated and unregulated flows. Moreover, we consider a loss function, $g(u_C, w_U)$, related to flooding, different from the (loss) objective function related to normal river operations (e.g. J_E in Eq. 3.7).

Under normal river conditions, water level in a storage y usually stays close to a reference value y_r . The water level is a measure of how severe a flood is, and the amount of damage caused by the flood increases with the water level. So for flooding, the water level itself can be considered as a suitable loss function. We assume the uncertainty w_U is governed by a probability measure \mathbb{P}_W on W , and the loss distribution can then be described from Eq. 4.3

$$\psi(u_C, \zeta) = \mathbb{P}_W \{w_U \in W | y(u_C, w_U) \leq \zeta\}, \quad \forall u_C \in U, \quad (5.1)$$

where y quantifies the loss. $\psi(u_C, \zeta)$ is described as the *loss distribution*, as it defines the probability measure on the sets inside W , where the loss, water level $y(u_C, w_U)$, is less than and equal to ζ .

As mentioned earlier, we use Value-at-Risk (VaR, Section 4.1.2) as a flood risk measure. A constraint on VaR associated with flooding, using water level as the loss function, can be obtained from Eq. 4.7,

$$\mathbb{P}_W \{w_U \in W : y(u_C, w_U) \leq y_{FL}\} \geq 1 - \epsilon_f, \quad (5.2)$$

where y_{FL} is a critical flood limit that should not be crossed with a high probability $1 - \epsilon_f$, where $\epsilon_f \in (0, 1)$ is a small value (e.g. 10^{-3}). If y crosses y_{FL} , then the flood risk is larger than what we are willing to accept. However, y_{FL} should be selected carefully, and it should not be too conservative to affect the river during normal conditions.

We use VaR as a flood risk measure, rather than CVaR (Section 4.1.2). Note that if the economic cost of the flood is strictly increasing with the maximum water level, then there is no difference between using a VaR constraint on the water level and a VaR constraint on the economic cost. For rivers this is often natural, since “the damage has been done”, once the river exceeds the rivers banks or reaches a particular flood level, and the VaR constraint will ensure that such an event has a very small probability.

5.1.2 Stochastic MPC problem for flood avoidance

In this section we use the river control problem formulation developed in Chapter 3 (Problem (3.55)), and extend it to include flood mitigation, leading to a Multiple Chance-Constrained optimisation Problem (M-CCP), which can be solved in a Stochastic MPC setting. For ease of reading, we restate Problem (3.55) below,

$$\begin{aligned} & \min_{\Theta_n, \Gamma_n} J(\Theta_n, \Gamma_n), \\ \text{s.t. } & \mathbb{P}\{\mathbf{w}_{U,n} \in \mathbb{W} : \mathbf{u}_{\min} \leq \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \mathbf{u}_{\max} \cap \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \Delta \mathbf{u}_{\max} \\ & \cap y_{\text{ref}} - y_m \leq \|\mathbf{C}\mathbf{x}_n(\mathbf{w}_{U,n})\|_{\infty} \leq y_{\text{ref}} + y_m\} \leq 1 - \epsilon, \end{aligned}$$

where J is the objective function given in Eqs. 3.52 & 3.53, the states dynamics \mathbf{x}_n are given in Eq. 3.37 and the control policy $\mathbf{u}_{C,n}$ is given in Eq. 3.38. Θ_n and Γ_n are the optimisation variables (see Eq. 3.38). $\mathbf{x}_n = [x_n \ \dots \ x_{n+M-1}]^T$, $\mathbf{u}_{C,n} = [u_{C,n} \ \dots \ u_{C,n+M-1}]^T$ and $\mathbf{w}_{U,n} = [w_{U,n} \ \dots \ w_{U,n+M-1}]^T$, where n is the time index, M is the horizon length and $\|\cdot\|_{\infty}$ is the infinity norm over the finite horizon, which gives the maximum value in the vector. Let $\mathbb{W} = W^M$, and let \mathbb{P} be the product measure (\mathbb{P}_w^M) over \mathbb{W} . For more details, see Section 3.4.

For compactness, we let $\bar{y}_n = \|\mathbf{C}\mathbf{x}_n(\mathbf{w}_{U,n})\|_{\infty}$, and let $y_{\text{ref}} - y_m = y_{\text{LL}}$ (Lower Limit) and $y_{\text{ref}} + y_m = y_{\text{UL}}$ (Upper Limit). With these notations, the above problem and the constraint on VaR (Eq. 5.2) can be represented as,

$$\begin{aligned} & \min_{\Theta_n, \Gamma_n} J(\Theta_n, \Gamma_n), \tag{5.3} \\ \text{s.t. } & \mathbb{P}\{\mathbf{w}_{U,n} \in \mathbb{W} : \mathbf{u}_{\min} \leq \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \mathbf{u}_{\max} \cap \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \Delta \mathbf{u}_{\max} \\ & \cap y_{\text{LL}} \leq \bar{y}_n(\mathbf{u}_{C,n}, \mathbf{w}_{U,n}) \leq y_{\text{UL}}\} \leq 1 - \epsilon. \\ & \mathbb{P}\{\mathbf{w}_{U,n} \in \mathbb{W} : \bar{y}_n(\mathbf{u}_{C,n}, \mathbf{w}_{U,n}) \leq y_{\text{FL}}\} \geq 1 - \epsilon_f, \end{aligned}$$

subject to the states dynamics, \mathbf{x}_n in Eq. 3.37, and the control policy, $\mathbf{u}_{C,n}$ in Eq. 3.38. $\epsilon \in (0, 1)$ and $\epsilon_f \in (0, 1)$ are user chosen probability levels, y_{LL} and y_{UL} are the recommended lower and upper limits on the water level respectively, which can be occasionally violated

with probability ϵ , without much harm. y_{FL} is the critical flood level limit that should only be exceeded with a very low probability $\epsilon_f \ll \epsilon$ over the time horizon.

Problem (5.3) is an M-CCP with two chance-constraints. The first chance-constraint is associated with normal operations, and the second chance-constraint is the VaR constraint related to flood risk mitigation. Problem (5.3) is non-convex in general, even if the constraints: ' $\mathbf{u}_{\min} \leq \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \mathbf{u}_{\max}$ ', ' $\Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}_{C,n}(\mathbf{w}_{U,n}) \leq \Delta \mathbf{u}_{\max}$ ', ' $y_{LL} \leq \bar{y}_n(\mathbf{u}_{C,n}, \mathbf{w}_{U,n}) \leq y_{UL}$ ' and ' $\bar{y}_n(\mathbf{u}_{C,n}, \mathbf{w}_{U,n}) \leq y_{FL}$ ' are convex. The problem is in the same form as Problem (4.14), stated in Chapter 4, and the algorithm proposed in Section 4.3.2 to find an approximate solution to Problem (4.14), is applicable to Problem (5.3). In the next section we will use this algorithm for flood mitigation in the upper part of Murray River in simulations.

The probabilities in Problem (5.3) are associated with the flow forecasts, and the quantiles of the distributions, particularly the very high ones are only approximately known, even in cases when plenty of the relevant historical data are available. Moreover, ϵ and ϵ_f in Problem (5.3), give the probability of violation over the time horizon used in the S-MPC problem, whereas in practice, one would be more interested in the probability over a much larger period. Due to the dependency between the S-MPC problems solved at each time instant, there is no easy way to infer such information e.g. a bound on the probability of a flood over a five years period. So, to some extent, ϵ and ϵ_f will be tuning parameters which can be adjusted based upon the experience of river operators.

5.2 Application of the proposed algorithm to the upper part of Murray River

In this section we apply the *Optimisation and Testing algorithm*, proposed in Section 4.3.2, to the upper part of Murray River in Australia, using historical data. For ease, we represent the sketch of the upper part of Murray River in Fig. 5.1. The figure shows the river stretch from Hume reservoir to Lake Mulwala. In Section 3.6 we applied optimisation schemes, developed in Section 3.5, to the river stretch to achieve normal control objectives of the stretch. In Section 3.6.4, we found that the schemes did not show satis-

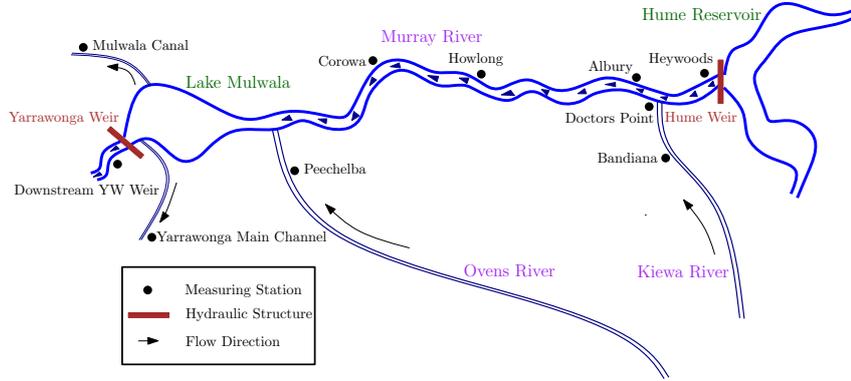


Figure 5.1: Hume Reservoir to Lake Mulwala on the Murray River (plot not to scale).

factory results when the unregulated inflows were high, because there was no flood risk mitigation strategy incorporated in the schemes. Now, we consider the flood avoidance Problem (5.3) for the same case, and apply the proposed algorithm (Section 4.3.2) to solve the problem and show that flood risk mitigation can be achieved while fulfilling normal control objectives.

5.2.1 Control design

To test the performance of the proposed algorithm, we picked the dataset with high inflows from the unregulated rivers (Kiewa and Ovens Rivers), as used in Section 3.6.4. Precisely, we used the dataset from 19th Sep., 2001 to 16th Nov., 2001, sampled at $T_s = 8$ hours. As before, we used a prediction horizon of $M = 20$ units in the MPC problem, which is roughly one week. We considered the following constraints to constitute the normal river operations requirement,

- $124.65 \leq y_{i,LM} \leq 124.9$, for $i = 1, 2, \dots, M$,
- $2,500 \leq Q_{i,H} \leq 30,000$, for $i = 1, 2, \dots, M$,
- $-500 \leq Q_{i,H} - Q_{i-1,H} \leq 1,200$, for $i = 1, 2, \dots, M$.

These constraints are required to be satisfied with a probability at least $1 - \epsilon = 1 - 0.1 = 0.9$. Apart from the above constraints, we considered the following constraint to constitute the flood operations requirement, and it is required to be satisfied with a probability

at least $1 - \epsilon_f = 1 - 0.01 = 0.99$,

- $y_{i,LM} \leq 125$, for $i = 1, 2, \dots, M$,

where ϵ and ϵ_f refer to the allowed violation probabilities in Problem (5.3). The above constraint forms the VaR constraint in the current problem. We set the *default solution* (Section 4.2.1) to be $Q_{H,0} = 1,000$ ML/Day, which is non-zero, because some release is necessary for riparian and in-stream environmental needs.

For ease of reading, we briefly restate the control design parameters set in Section 3.6.1. We selected the matrices \mathbf{Q} and \mathbf{S} in the objective function $J(\Theta_n, \Gamma_n)$ (Eq. 3.52) as block diagonal matrices with M copies of Q' and S' matrices, where

$$Q' = \begin{bmatrix} 100 & \mathbf{0}_{1, n_s-1} \\ \mathbf{0}_{n_s-1, 1} & \mathbf{0}_{n_s-1, n_s-1} \end{bmatrix} \quad S' = 10^{-11},$$

$n_s = 21$ is the number of states, and the first state corresponded to the water level in Lake Mulwala (Appendix 3.C). We used $\mathbf{R} = \mathbf{0}$ (in Eq. 3.52), since the change in flow at Heywoods is already subjected to constraints. We used the parametrisation of Γ_n and Θ_n matrices (in Eq. 3.38) as in Eq. 3.19, to keep the number of optimisation variables d small. We got $d = 78$ decision variables in the optimisation Problem (5.3), for details see Section 3.6.1. We selected the overall POF (Probability Of Failure) of the algorithm to be $\delta = 1 \times 10^{-6}$, which makes $\beta = \frac{1}{7} \times 10^{-6}$, and we fixed the safety margin $\varrho = \frac{\epsilon_f}{2} = 0.005$. For simulations, we selected the following two situations:

- Situation 1: We considered the unregulated inflows from Kiewa and Ovens River to be unknown, however, we assumed the demands from irrigation channels: Yarrowonga Main Channel (YMC) and Mulwala Canal (MC) and from Downstream Yarrowonga Weir (DYW) to be exactly known. Forecasts for the flows at Bandiana (B) and Peechelba (P) are based on Eq. 3.39, and we used the forecast models obtained in Appendices 2.C & 2.D. We selected the uncertainties in the flow forecasts w_B and w_P as independent and identically distributed Gaussian random variables with zero means, and 50 ML/Day and 200 ML/Day standard deviations respectively.

- Situation 2: We considered the unregulated in- and out-flows to be partially known, except the release from Yarrawonga Weir, which was known completely. Flow forecasts at Bandiana and Peechelba were done the same way as in Situation 1. However, the demands to the irrigation channels were considered to be known for the first ten steps of the horizon $M = 20$, and for the next ten steps, we considered the average value of the actual demands in those 10 steps. Such a demand profile is close to reality, because river operators are usually aware of the demands for 3 to 4 days in future.

The optimisation problems in the algorithm were solved by running YALMIP [115] over SDPT3 [4].

5.2.2 Performance of Situation 1: (unknown unregulated inflows)

For ease of reading, we show the flows at Heywoods (H), Bandiana (B), Peechelba (P), Downstream Yarrawonga Weir (DYW), Yarrawonga Main Channel (YMC) and Mulwala Canal (MC), once again in Fig. 5.2. Fig. 5.3 shows the simulations of the flows at Bandiana and Peechelba for the selected dataset. The blue curves show the actual recorded data and the red dashed curves show the simulation of the forecast models. The simulation results of the flows at Bandiana are good, however, for the flows at Peechelba, the model picks the trends well, but at flow peaks, the simulated flows are off by several hundred ML/Day from the actual flows. For this reason we selected relatively higher standard deviation of the additive noise term w_p .

Fig. 5.4 shows the controlled water level in Lake Mulwala. The blue curve in the figure shows the controlled water level, the black curve shows the actual recorded water level and the magenta curve shows the simulation of the model in Eq. 3.65, using the measured input data. The model performed reasonably well and picked the main trends in the selected dataset.

The water level was controlled well within 124.65 and 124.9 mAHD through out the simulation (Fig. 5.4). However, it was about to hit the boundary (124.9 mAHD) at the 42nd time instant. It can be explained from the regulated flows at Heywoods. In Fig. 5.5, the red curve shows the actual recorded flow release at Heywoods and the blue curve

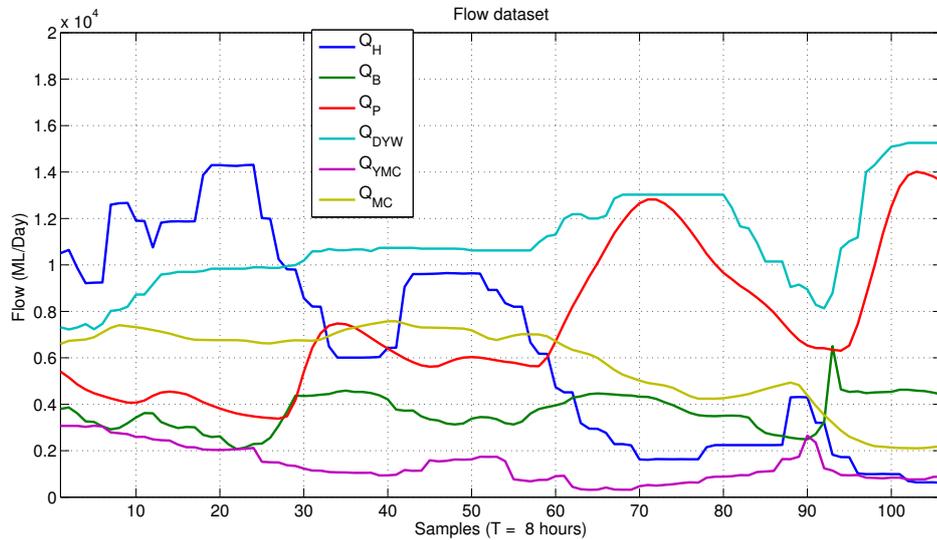


Figure 5.2: Inflows and outflows in the dataset.

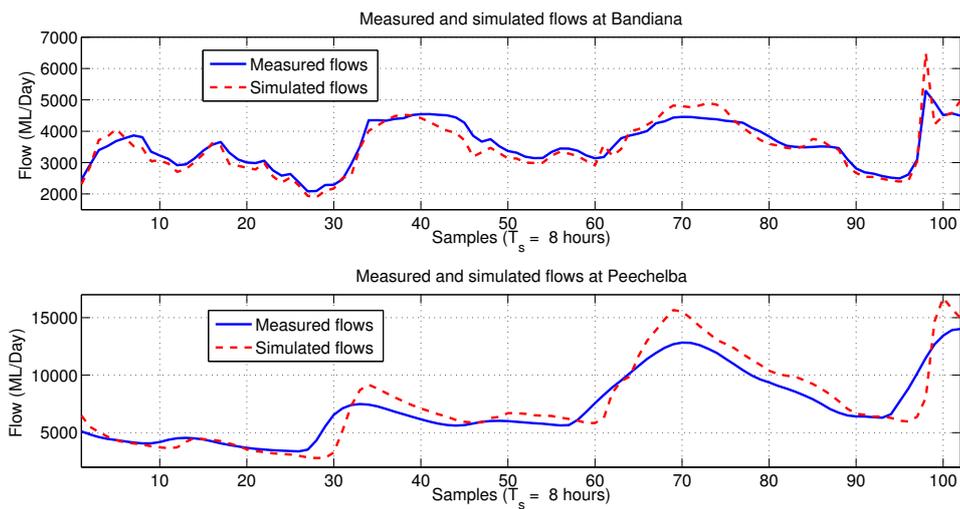


Figure 5.3: Measured and simulated flows at Bandiana and Peechelba.

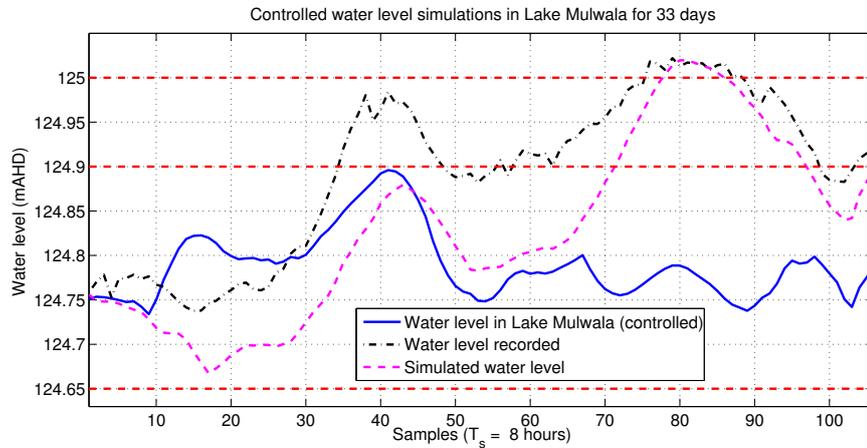


Figure 5.4: Controlled water level in Lake Mulwala.

shows the flow release obtained from the proposed strategy. From the sampling instant 22^{nd} to 37^{th} , the flows were decreased at the maximum allowed rate, i.e. -500 ML/Day. The restriction, on the rate of flow decrease, caused the water level to rise in the lake as shown in Fig. 5.4. In Fig. 5.5, the flows at Heywoods went below $2,500$ ML/Day twice ($2,500$ ML/Day corresponds to the upper black dashed curve in the figure). These were the events when the test in Step B-3 of the algorithm (Section 4.3.2), with $\epsilon_V = \epsilon_f$, failed. In the test, the water level crossed 125 mAHD more than 44 times out of $N_T = 8,798$ different noise scenarios, and the improvement procedure was called. The value $N_T = 8,798$ was obtained from the procedure $\text{Find_N_Test}(\epsilon_f = 0.01, \rho = 0.005, \beta_T = \frac{1 \times 10^{-6}}{7})$ (for details, see Section 4.3.1), and the number 44 was obtained from $(\epsilon_f - \rho)N_T$. The flows were decreased up to $1,000$ ML/Day (*default solution*, $Q_{H,0}$) at those events, as shown in Fig. 5.5 ($1,000$ ML/Day corresponds to the lower black dashed curve in the figure).

Fig. 5.6 shows the α values obtained from Problem (4.26) in the algorithm. A non-zero α value indicates an event, where the *improving* phase of the algorithm was called, and $\alpha = 1$ indicates the instants when the flows were reduced to $1,000$ ML/Day, i.e. the *default solution* was used. Fig. 5.6 shows that the *improving* routine has been called in roughly 30% of the total simulation steps. It shows that for the rest of the time, there was no need to solve Problem (4.26), and a solution of Problem (5.3) was available at the end of the *optimisation* phase of the algorithm.

Fig. 5.5 shows a couple of non-smooth changes in the regulated flow release from

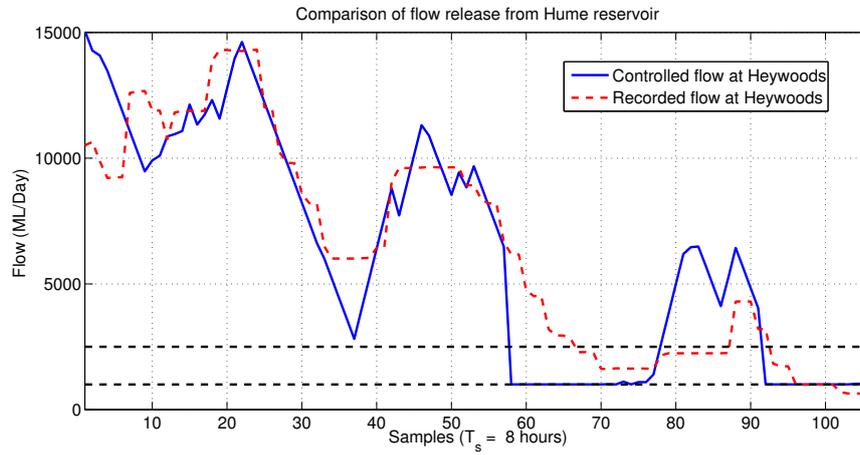
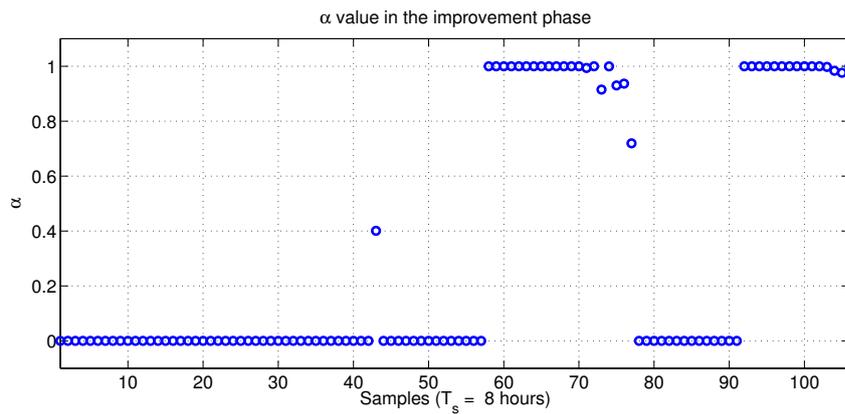


Figure 5.5: Regulated flows at Heywoods.

Figure 5.6: α values obtained in the *improving* phase (Problem (4.26)).

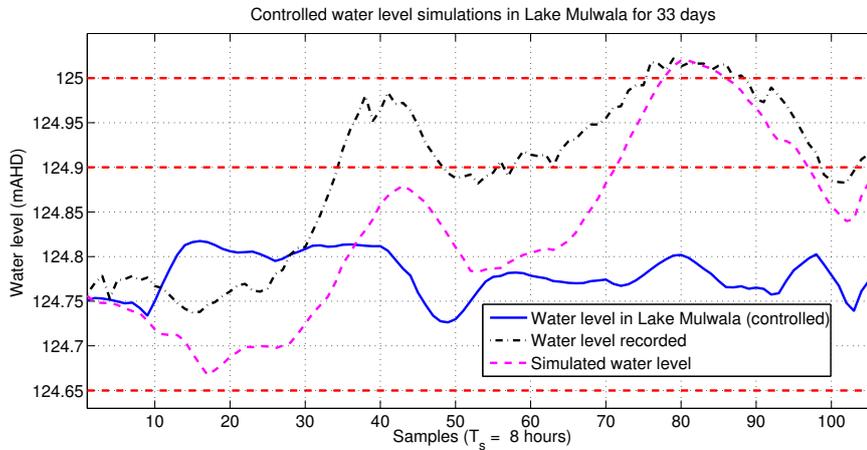


Figure 5.7: Controlled water level in Lake Mulwala with non-zero \mathbf{R} in the objective function.

Hume Reservoir. This can be smoothed down by considering a non zero \mathbf{R} matrix in the objective criterion $J(\Theta_n, \Gamma_n)$ (Eq. 3.52). For this, we selected the \mathbf{R} matrix as an $M \times M$ diagonal matrix with $R' = 5 \times 10^{-8}$ in the diagonal. Figs. 5.7 & 5.8 show the corresponding results. Fig. 5.8 shows that with this change the control action changed smoothly, as compared to what we obtained in Fig. 5.5. In this case, we did not experience any rise in the water level close to the 42nd time instant, as we saw before in Fig. 5.4. The explanation can be seen in Fig. 5.8, where the flows did not get very high around the 22nd sampling instant, as compared to the flow behaviour in Fig. 5.5, where it got really high (up to 15,000 ML/Day) around the 22nd sampling instant. The performance of the algorithm for the rest of the time was almost similar in the two cases, and the water level was well maintained between its upper and lower limits, as shown in Figs. 5.4 & 5.7. Fig. 5.9 shows the corresponding α values obtained from Problem (4.26), which indicates that the number of calls to the *improving* phase was almost similar to what it was in the previous case (Fig. 5.6), where \mathbf{R} was selected as $\mathbf{0}$.

In this section we have seen that with the application of *Optimisation and Testing algorithm*, we managed to keep the water level within safe limits, even when we have large unregulated inflows. As mentioned earlier, the comparison with the recorded data is not completely fair, because we had access to the exact future water demands and we adjusted the flow release every 8 hours while the operators only adjusted the flow every 24

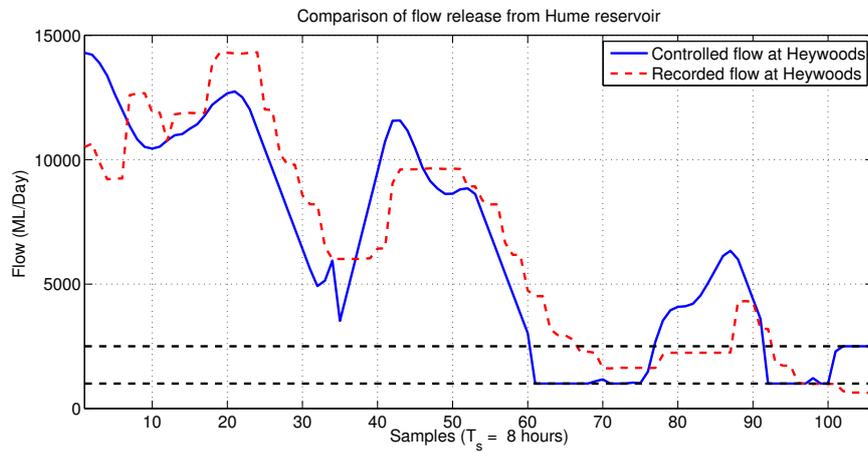


Figure 5.8: Regulated flows at Heywoods with non-zero \mathbf{R} in the objective function.

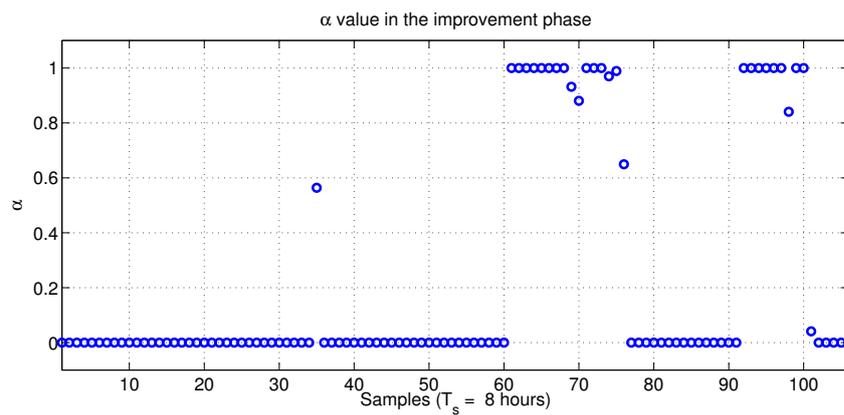


Figure 5.9: α values obtained in the *improving* phase (Problem (4.26)), with non-zero \mathbf{R} in the objective function.

hours. To see the response of the algorithm in a more realistic setting, we repeated the same simulations in the next section, with partially known future demands

5.2.3 Performance of Situation 2: (partially known future water demands)

As we did in Section 3.6.3, here also we consider the case when water demands from the irrigation channels (Yarrowonga Main Channel and Mulwala Canal) are unknown. We mentioned in Section 3.6.3 that the river operators mostly know the water demands for 3 to 4 days in future. So in the following simulations, we considered the case where we knew the water demands for the first 10 steps (roughly 3.5 days) of the finite horizon $M = 20$, and for the next 10 steps, we assumed the demands to be the average value of the demands over those 10 steps. Also, in these simulations we have used a non zero \mathbf{R} matrix (in Eq. 3.52), and we selected it as an $M \times M$ diagonal matrix with $R' = 5 \times 10^{-8}$ on the diagonal.

Figs. 5.10 & 5.11 show the corresponding control results obtained by using the *Optimisation and Testing algorithm*. As expected, they are similar to the previous simulation results, because in this particular dataset, the inflows from the unregulated rivers are large (see Fig. 5.2), and in such cases the uncertainty in the inflows forecast is expected to be higher than the uncertainty in the future demands. This effect dominated the simulation results as well, and the uncertainties in the future demands stayed hidden.

The simulation results can be explained the same way as we did before. Fig. 5.10 shows the water level in Lake Mulwala, and again the control algorithm kept the water level (shown with the blue curve) between 124.65 and 124.9 mAHD, through out the simulations. However, the actual water level (shown with the black dashed-dotted curve) exceeded the upper limit multiple times, and it also crossed the flood limit (125 mAHD). As observed earlier, the water level close to the 45th time instant got closer to the upper limit, and this was due to the high inflow from Peechelba at around 33rd time instant, shown in Fig. 5.2. To compensate for that, the algorithm reduced the flow release from Hume Reservoir from 20th to 36th time instant, shown by the blue curve in Fig. 5.11. During this period, the flow was reduced at the maximum allowed rate (i.e. -500 ML/Day), and it could not go below that, which caused the water level in the lake to rise.

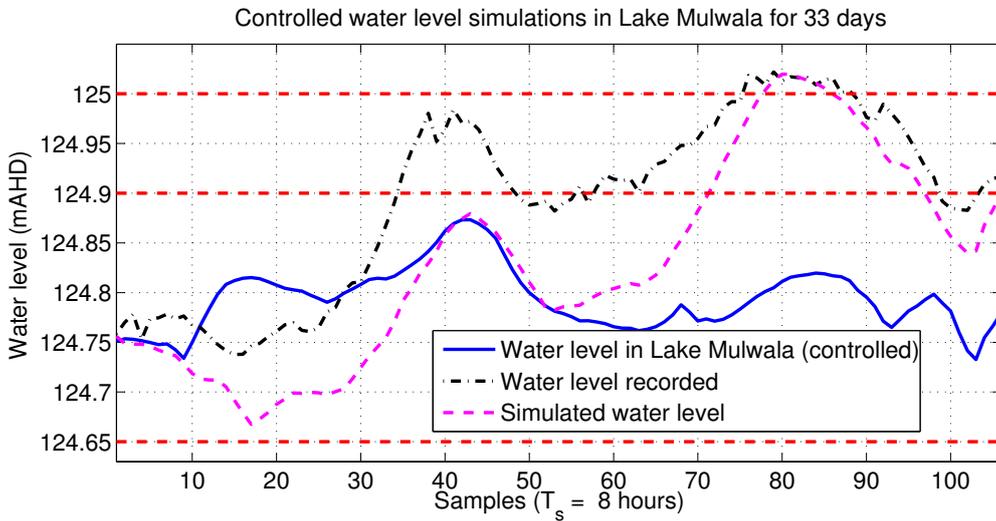


Figure 5.10: Controlled water level in Lake Mulwala.

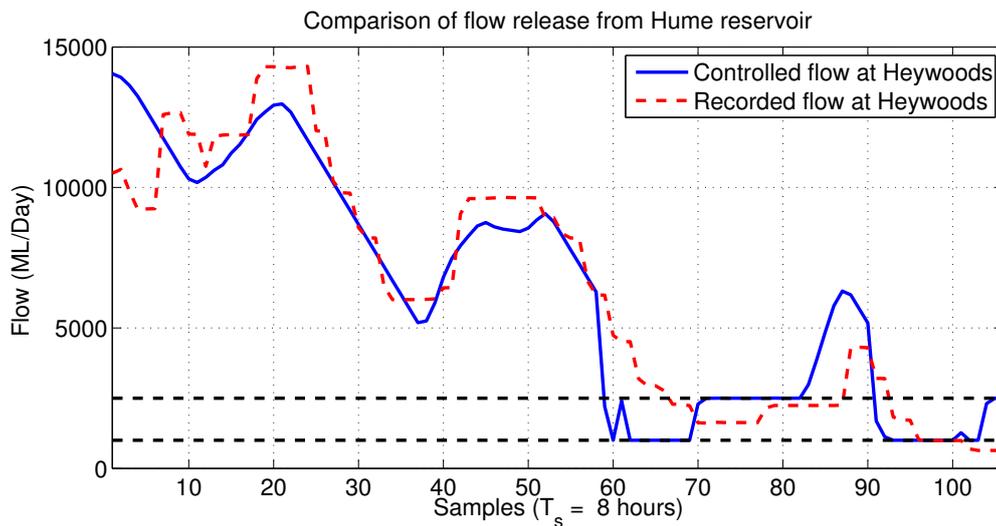


Figure 5.11: Regulated flows at Heywoods.

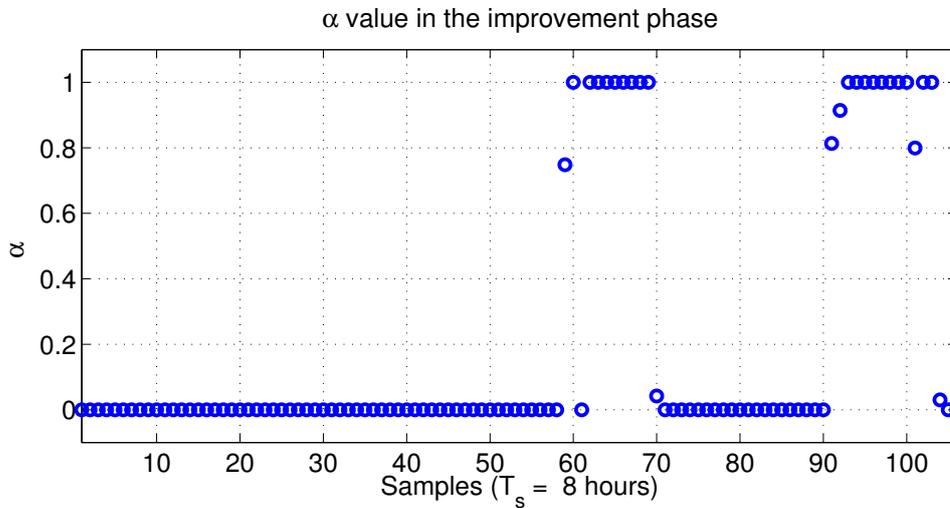


Figure 5.12: α values obtained in the *improving* phase (Problem (4.26)).

Fig. 5.11 shows that the flows were reduced up to the *default solution* level (1,000 ML/Day) multiple times, in the second half of the simulations. At those instants, the chance-constraint corresponding to the flood limit failed the test, and again, it was due to the high inflows from the unregulated rivers (Kiewa and Ovens Rivers), shown in Fig. 5.2. Timely reduction in the flow release from Hume Reservoir prevented the water level in Lake Mulwala to cross the upper limit, and this is what we have seen in the previous simulations as well. Fig. 5.12 shows the corresponding α values, used by the algorithm in each MPC step. Again, a zero value corresponds to the events when the *improving* phase was not called, and the algorithm called the *improving* phase only 30-40 percent times during the whole simulation, which is computationally very favourable.

The simulation results confirm that formulating the flood mitigation problem as an M-CCP with two chance constraints (Problem (5.3)) achieves normal control objectives, during normal conditions, and avoids flooding whenever flood risks appear, with an aid of the *default solution*. Also, the proposed algorithm (Section 4.3.2) finds approximate solution of the M-CCP problem in a computationally affordable way.

Chapter 6

Conclusions and Future Works

6.1 Conclusions

The main objective of this thesis is to model and control rivers. We prefer data-based modelling due to the availability of operational data from rivers, and also the obtained models are simple and easy to use for control purposes. The aim of the control part is to design river control, using the developed models, which should perform both normal river operations and flood avoidance.

In Chapter 2 we have used the upper part of Murray River in Australia as a case study, and compared different system identification methods. Particularly, in this thesis, we modelled water level in Lake Mulwala and flows at Doctors Point, Bandiana and Peechelba, which are the important variables of the river stretch and the catchment. The models were identified and validated against different data. In most cases the models performed well, and they were found suitable for control and simulation purposes. In particular, optimisation based methods, where prior information can easily be incorporated, performed very well on the river example.

To effectively control a river, we need to consider all regulated and unregulated in- and out-flows in the river stretch. Moreover, forecasts of the unregulated in- and out-flows are also required. These forecasts carry uncertainties, which we also want to accommodate in the control strategy. In Chapter 3 we showed that Stochastic Model Predictive Control (S-MPC) is a promising control strategy, since it can accommodate forecasts. Moreover, MPC in general, is a suitable technique to incorporate constraints, and rivers have many physical, environmental and operational constraints. However, due to the

presence of the uncertainties, constraints are not always satisfied and the feasibility of the river control (optimisation) problem is not guaranteed. To compensate for that, the river control problem is formulated as a Chance-Constrained optimisation Problem (CCP), to be solved in an S-MPC setup. A CCP consists of a probabilistic constraint, which is non-convex in the optimisation variables in general, and we use the scenario approach to find approximate solutions of CCPs. A randomised version of such an S-MPC is then used to propose two optimisation schemes that provide feasibility assurance of the river control problem. The first scheme requires a user specified trade-off between performance (in terms of meeting control objectives) and constraint relaxation, while the second scheme is an automated option. Both schemes were successfully applied in simulations to control the water level in Lake Mulwala on Murray River.

To further improve the river control, so that it can also incorporate a flood mitigation strategy in the developed control schemes, we proposed Value-at-Risk as a flood risk measure, and incorporated it in the river control formulation as a second chance-constraint. This leads to a Multiple Chance-Constrained optimisation Problem (M-CCP) with two chance-constraints. Chapter 4 is dedicated to find solutions of M-CCPs with two or any finite number of chance-constraints, in a computationally affordable way within an S-MPC setting. We proposed *Optimisation and Testing algorithm* to find approximate solutions of such M-CCPs.

The proposed algorithm is applicable to river control problems, which ensure flood avoidance without leading to conservative results during normal conditions. The algorithm is applied, in simulations, to control the water level in Lake Mulwala on Murray River with high unregulated inflows in Chapter 5. The comparison of the results with the historical data and with the control strategies, where flood mitigation was not considered, shows a clear improvement in performance of the proposed strategy.

6.2 Future works

The work in this thesis can be extended in several directions. A few of them are given below,

1. Phenomena like rainfall-runoff, evaporation and ground water-surface water interaction can be incorporated in river models, provided the relevant measurements are available. Such models can be very useful for simulation and forecasting purposes, and it will be interesting to see how much improvement they can provide for control purposes, if compared with the performance of the river models presented in this thesis.
2. Unregulated tributaries can be modelled in a more sophisticated way, where they might also require incorporation of phenomena like rainfall-runoff, evaporation and ground water-surface water interaction. Moreover, it will be ideal if the models could cover a wide range of flows. For the upper part of Murray River, it is important to have a better forecast model for Ovens River, because it has contributed to many floods in the past
3. Implementation of the proposed S-MPC strategy in Chapters 3 & 5 to an actual river can be pursued, e.g. the designed controllers in this thesis can be applied to the upper part of Murray River.
4. The river stretch considered in this thesis consists of a single long reach. It will be useful to see the performance of the proposed schemes on multiple reaches of a river, e.g. the stretch between Yarrawonga Weir and Turrumbarry Weir (which is further downstream, along the Murray River) can also be included in the control design.
5. More sophisticated environmental aspects can be considered in the modelling and control procedures, e.g. ensuring reduction in channel erosion, safe fish habitat and avoidance of sedimentation and bank degradation.
6. The proposed *Optimisation and Testing algorithm* can be further improved in terms of computational cost of finding approximate solutions of general M-CCPs.

This page intentionally left blank.

Bibliography

- [1] P.C. Young. The CAPTAIN toolbox for Matlab. In *IFAC Symposium on System Identification*, volume 14, pages 909–914, 2006.
- [2] H. Garnier and M Gilson. The CONTSID toolbox for Matlab, Centre de recherche en Automatique de Nancy (CRAN), University of Lorraine, France. www.cran.uhp-nancy.fr/contsid.
- [3] J. Lofberg. YALMIP: A toolbox for modeling and optimization in MATLAB. In *International Symposium on Computer Aided Control Systems Design*, pages 284–289. IEEE, 2004.
- [4] K.C. Toh, M.J. Todd, and R.H. Tütüncü. SDPT3 — a MATLAB software package for semidefinite programming, version 1.3. *Optimization methods and software*, 11(1-4):545–581, 1999.
- [5] 2030 Water Resources Group, <http://www.2030wrg.org>, 2008.
- [6] 2030 WRG. The water resources group - background, impact and the way forward. In *World Economic Forum Annual Meeting 2012*.
- [7] UNESCO. World water assessment program: water - a shared responsibility. *The United Nations World Water Development Report*, 2006.
- [8] M. Foo. PhD Thesis: Modelling and control design of river systems. 2011.
- [9] M.H. Chaudhry. *Open-channel flow*. Springer Publishing Company, 2007.
- [10] R. Szymkiewicz. *Numerical Modeling in Open Channel Hydraulics*. Springer, 2010.

-
- [11] M.G. Bos. Discharge measurement structures. *International Institute for Land Reclamation and Improvement*, 1989.
- [12] M. Foo, S.K. Ooi, and E. Weyer. Modelling of rivers for control design. In *System Identification, Environmental Modelling, and Control System Design*, pages 423–447, 2012.
- [13] M. Maxwell and S. Warnick. Modeling and identification of the Sevier River system. In *IEEE American Control Conference*, 2006.
- [14] F. Pianosi, A. Castelletti, and M. Lovera. Identification of a flow routing model for the Red River network. In *IFAC Symposium on System Identification*, volume 16, pages 1037–1042, 2012.
- [15] B. Sohlberg and M. Sernfalt. Grey box modelling for river control. *Journal of Hydroinformatics*, 4:265–280, 2002.
- [16] P.C. Young. Top-down and data-based mechanistic modelling of rainfall-flow dynamics at the catchment scale. *Hydrological processes*, 17(11):2195–2217, 2003.
- [17] P.C. Young, A. Castelletti, and F. Pianosi. The data-based mechanistic approach in hydrological modelling. *Topics on System Analysis and Integrated Water Resource Management*, pages 27–48, 2007.
- [18] L. Ljung. System identification: Theory for the user. *PTR Prentice Hall, Upper Saddle River, NJ*, 1999.
- [19] M. Verhaegen and V. Verdult. *Filtering and System Identification: a Least Squares approach*. Cambridge University Press, 2007.
- [20] D.F. Lekkas and C. Onof. Introducing the variation of advective time delay (ATD) to transfer function models. *Environmental Modelling & Software*, 21(8):1180–1189, 2006.
- [21] C.E. Imrie, S. Durucan, and A. Korre. River flow prediction using artificial neural networks: generalisation beyond the calibration range. *Journal of Hydrology*, 233(1):138–153, 2000.

- [22] J.M. Maciejowski. *Predictive control with constraints*. Pearson education, 2002.
- [23] I. Batina. *Model predictive control for stochastic systems by randomized algorithms*. PhD thesis, Technische Universiteit Eindhoven, 2004.
- [24] M. Cannon, B. Kouvaritakis, and X. Wu. Probabilistic constrained MPC for multiplicative and additive stochastic uncertainty. *IEEE Transactions on Automatic Control*, 54(7):1626–1632, 2009.
- [25] E. Cinquemani, M. Agarwal, D. Chatterjee, and J. Lygeros. Convexity and convex approximations of discrete-time stochastic control problems with constraints. *Automatica*, 47(9):2082–2087, 2011.
- [26] J.A. Primbs and C.H. Sung. Stochastic receding horizon control of constrained linear systems with state and control multiplicative noise. *Automatic Control, IEEE Transactions on*, 54(2):221–230, 2009.
- [27] M. Prandini, S. Garatti, and J. Lygeros. A randomized approach to stochastic model predictive control. In *51st Conference on Decision and Control (CDC)*, pages 7315–7320. IEEE, 2012.
- [28] G.C. Calafiore and M.C. Campi. The scenario approach to robust control design. *IEEE Transactions on Automatic Control*, 51(5):742–753, 2006.
- [29] M.C. Campi and S. Garatti. A sampling-and-discarding approach to chance-constrained optimization: feasibility and optimality. *Journal of Optimization Theory and Applications*, 148(2):257–280, 2011.
- [30] M.C. Campi, S. Garatti, and M. Prandini. The scenario approach for systems and control design. *Annual Reviews in Control*, 33(2):149–157, 2009.
- [31] J.A. Cunge, F.M. Holly, and A. Verwey. Practical aspects of computational river hydraulics. 1980.
- [32] A.O. Akan. *Open channel hydraulics*. Butterworth-Heinemann, 2011.

- [33] M. Papageorgiou and A. Messmer. Continuous-time and discrete-time design of water flow and water level regulators. *Automatica*, 21(6):649–661, 1985.
- [34] J. Schuurmans, O. Bosgra, and R. Brouwer. Open-channel flow model approximation for controller design. *Applied Mathematical Modelling*, 19(9):525–530, 1995.
- [35] P.J. Van Overloop, J. Schuurmans, R. Brouwer, and C. M. Burt. Multiple-model optimization of Proportional Integral controllers on canals. *Journal of irrigation and drainage engineering*, 131(2):190–196, 2005.
- [36] L.G. Silvis, A. Hof, P.M.J van den Hof, and A.J. Clemmens. System identification on open channels. In *3rd Proceedings of International Conference on Hydroinformatics.*, pages 21–224. Balkema, Rotterdam, The Netherlands, 1998.
- [37] P-O. Malaterre and J-P. Baume. Modeling and regulation of irrigation canals: existing applications and ongoing researches. In *IEEE International Conference on Systems, Man, and Cybernetics, 1998*, volume 4, pages 3850–3855.
- [38] J.M. Coron, B. DAndréa-Novel, and G. Bastin. A Lyapunov approach to control irrigation canals modeled by Saint-Venant equations. In *Proceeding of European Control Conference, Karlsruhe, 1999*.
- [39] G. Bastin, J.M Coron, B. D’Andréa-Novel, et al. On Lyapunov stability of linearised Saint-Venant equations for a sloping channel. *Networks and Hetrogeneous Media*, 4(2):177–187, 2009.
- [40] A. Diagne, G. Bastin, and J.M. Coron. Lyapunov exponential stability of 1-D linear hyperbolic systems of balance laws. *Automatica*, 48(1):109–114, 2012.
- [41] X. Litrico and V. Fromion. *Modeling and control of hydrosystems*. Springer Science & Business Media, 2009.
- [42] X. Litrico and D. Georges. Nonlinear identification of an irrigation system. In *Proceedings of the 36th IEEE Conference on Decision and Control.*, volume 1, pages 852–857. IEEE, 1997.

- [43] X. Litrico and D. Georges. Robust continuous-time and discrete-time flow control of a dam–river system (I) Modelling. *Applied mathematical modelling*, 23(11):809–827, 1999.
- [44] X. Litrico and V. Fromion. Analytical approximation of open-channel flow for controller design. *Applied Mathematical Modelling*, 28(7):677–695, 2004.
- [45] X. Litrico and V. Fromion. Simplified modeling of irrigation canals for controller design. *Journal of Irrigation and Drainage Engineering*, 130(5):373–383, 2004.
- [46] E. Weyer. System identification of an open water channel. *Control engineering practice*, 9(12):1289–1299, 2001.
- [47] K. Eurén and E. Weyer. System identification of open water channels with under-shot and overshoot gates. *Control Engineering Practice*, 15(7):813–824, 2007.
- [48] E. Weyer. Control of irrigation channels. *IEEE Transactions on Control Systems Technology*, 16(4):664–675, 2008.
- [49] M. Cantoni, E. Weyer, Y. Li, S.K. Ooi, I. Mareels, and M. Ryan. Control of large-scale irrigation networks. *Proceedings of the IEEE*, 95(1):75–91, 2007.
- [50] S.K. Ooi and E. Weyer. Control design for an irrigation channel from physical data. *Control Engineering Practice*, 16(9):1132–1150, 2008.
- [51] M. Foo, S.K. Ooi, and E. Weyer. System identification and control of the Broken River. *IEEE Transactions on Control Systems Technology*, 2013.
- [52] H. Elfawal-Mansour, D. Georges, and N. Ohnish. Optimal control of an open channel irrigation system based on nonlinear models. In *Proceedings of TENCON 2000.*, volume 3, pages 308–313. IEEE, 2000.
- [53] P.C. Young. Hypothetico-inductive data-based mechanistic modeling of hydrological systems. *Water Resources Research*, 49(2):915–935, 2013.
- [54] P. Young. The identification and estimation of nonlinear stochastic systems. In *Nonlinear dynamics and statistics*, pages 127–166. Springer, 2001.

- [55] A. Castelletti, F. Pianosi, R. Soncini-Sessa, and P. Young. A DBM model for snowmelt simulation. In *Proceedings of IFAC Symposium on System Identification, Saint-Malo, France, 2009*.
- [56] G. Bastin, L. Moens, P. Dierickx, et al. On line river flow forecasting with Hydro-max: successes and challenges after twelve years of experience. In *Proceedings of the 15th IFAC Symposium on System Identification, SYSID, Saint-Malo, France*, pages 1774–1779. Citeseer, 2009.
- [57] M. Papageorgiou and A. Messmer. Flow control of a long river stretch. *Automatica*, 25(2):177–183, 1989.
- [58] X. Litrico and D. Georges. Robust LQG control of single input multiple-outputs dam-river systems. *International Journal of Systems Science*, 32(6):795–805, 2001.
- [59] X. Litrico and V. Fromion. Real-time management of multi-reservoir hydraulic systems using H-infinity optimization. In *Proceedings of IFAC World Congress*, volume 15, pages 1409–1409, 2002.
- [60] G. Glanzmann, M. von Siebenthal, T. Geyer, G. Papafotiou, and M. Morari. Supervisory water level control for cascaded river power plants. In *Proceedings of the 6th International Conference on Hydropower, Stavanger, Norway*, pages 423–447, 2005.
- [61] C. Setz, Ad. Heinrich, P. Rostalski, G. Papafotiou, and M. Morari. Application of Model Predictive Control to a cascade of river power plants. In *Proceedings of the IFAC World Congress*, 2008.
- [62] V. Puig, J. Romera, J. Quevedo, C. Cardona, A. Salterain, E. Ayesa, I. Irizar, A. Castro, M. Lujan, P. Charbonnaud, et al. Optimal predictive control of water transport systems: Arrêt-darré/arrois case study. *Water Science & Technology WST*, 60(8):2125–2133, 2009.
- [63] P.J. Van Overloop, R.R. Negenborn, B. De Schutter, and N.C. Van De Giesen. Predictive control for national water flow optimization in the netherlands. In *Intelligent infrastructures*, pages 439–461. Springer, 2010.

- [64] J. Wagenpfeil, E. Arnold, and O. Sawodny. Modeling and optimized water management of inland waterway systems. In *IEEE International Conference on Control Applications (CCA)*, pages 1874–1880. IEEE, 2010.
- [65] H. Linke. A model predictive controller for optimal hydro-power utilization of river reservoirs. In *IEEE International Conference on Control Applications (CCA)*, pages 1868–1873. IEEE, 2010.
- [66] A. Şahin and M. Morari. Decentralized Model Predictive Control for a cascade of river power plants. In *Intelligent Infrastructures*, pages 463–485. Springer, 2010.
- [67] L. Raso. *Optimal control of water systems under forecast uncertainty: Robust, Proactive, and Integrated*. TU Delft, Delft University of Technology, 2013.
- [68] A. Ficchi, L. Raso, P.O. Malaterre, D. Dorchies, M. Jay-Allemand, F. Pianosi, P.J. van Overloop, and G. Thirel. Short term reservoirs operation on the Seine River: Performance analysis of Tree-Based Model Predictive Control. *International conference on Hydroinformatics*, 2014.
- [69] D.K. Delgoda, S.K. Saleem, M.N. Halgamuge, and H. Malano. Multiple model predictive flood control in regulated river systems with uncertain inflows. *Water resources management*, 27(3):765–790, 2013.
- [70] T.B. Blanco, P. Willems, Po-K. Chiang, N. Haverbeke, J. Berlamont, and B. De Moor. Flood regulation using nonlinear model predictive control. *Control Engineering Practice*, 18(10):1147–1157, 2010.
- [71] M. Breckpot, O.M. Agudelo, and B. De Moor. Flood control with model predictive control for river systems with water reservoirs. *Journal of Irrigation and Drainage Engineering*, 139(7):532–541, 2013.
- [72] M. Breckpot, O.M. Agudelo, P. Meert, P. Willems, and B. De Moor. Flood control of the Demer by using Model Predictive Control. *Control Engineering Practice*, 21(12):1776–1787, 2013.

- [73] P.K. Chiang and P. Willems. Combine evolutionary optimization with model predictive control in real-time flood control of a river system. *Water Resources Management*, 29(8):2527–2542, 2015.
- [74] L. Davis. Handbook of genetic algorithms. *Van Nostrand Reinhold, New York*, 1991.
- [75] M. Kearney, P. Dower, and M. Cantoni. Model predictive control for flood mitigation: A Wivenhoe Dam case study. In *Australian Control Conference (AUCC), 2011*, pages 290–296. IEEE, 2011.
- [76] publisher=International Water Association Romera, J. and Ocampo-Martinez, C. and Puig, V. and Quevedo, J. and García, P. and Pérez, G. and others, year=2011. Flooding management using hybrid model predictive control at the Ebro River in Spain.
- [77] H. van Ekeren, R.R. Negenborn, P.J. van Overloop, and B. De Schutter. Hybrid model predictive control using time-instant optimization for the Rhine-Meuse Delta. In *IEEE International Conference on Networking, Sensing and Control (ICNSC)*, pages 215–220. IEEE, 2011.
- [78] X. Tian, P.J. van Overloop, R.R. Negenborn, and N. van de Giesen. Operational flood control of a low-lying delta system using large time step Model Predictive Control. *Advances in Water Resources*, 75:1–13, 2015.
- [79] T. Söderström and P. Stoica. *System Identification*. Prentice-Hall, Inc., 1988.
- [80] P.C. Young, H. Garnier, and M. Gilson. Refined instrumental variable identification of continuous-time hybrid Box-Jenkins models. In *Identification of continuous-time models from sampled data*, pages 91–131. Springer, 2008.
- [81] H. Garnier, M. Mensler, and A. Richard. Continuous-time model identification from sampled data: implementation issues and performance evaluation. *International Journal of Control*, 76(13):1337–1357, 2003.

- [82] P. Patrinos, S. Trimboli, and A. Bemporad. Stochastic MPC for real-time market-based optimal power dispatch. In *Decision and Control and European Control Conference (CDC-ECC)*, pages 7111–7116. IEEE, 2011.
- [83] D.P. De Farias and B. Van Roy. On constraint sampling in the linear programming approach to approximate dynamic programming. *Mathematics of operations research*, 29(3):462–478, 2004.
- [84] E. Erdođan and G. Lyengar. Ambiguous chance constrained problems and robust optimization. *Mathematical Programming*, 107(1-2):37–61, 2006.
- [85] A. Nemirovski and A. Shapiro. Scenario approximations of chance constraints. In *Probabilistic and randomized methods for design under uncertainty*, pages 3–47. Springer, 2006.
- [86] R.T Rockafellar and St Uryasev. Conditional value-at-risk for general loss distributions. *Journal of Banking & Finance*, 26(7):1443–1471, 2002.
- [87] P.V. Overschee and B.D. Moor. *Subspace Identification for the Linear Systems: Theory and Implementation*. Boston: Kluwer Academic Publishers, 1996.
- [88] P.C. Young. Data-based mechanistic modelling of environmental systems. In *Proceedings of IFAC Workshop on Environmental Systems, First Plenary Session Keynote Paper, Yokohama, Japan*, 2001.
- [89] H.A. Nasir and E. Weyer. Estimation of models for the upper part of Murray River with flow dependent parameters. In *IFAC Symposium on System Identification*, 2015.
- [90] H.A. Nasir and E. Weyer. Comparison of prediction error methods and subspace identification methods for rivers. In *3rd Australian Control Conference (AUCC)*, pages 415–420. IEEE, 2013.
- [91] M. Thomassin, T. Bastogne, and A. Richard. Identification of a managed river reach by a Bayesian approach. *IEEE Transactions on Control Systems Technology*, 17(2):353–365, 2009.

- [92] S. Björklund. *A survey and comparison of time-delay estimation methods in linear systems*. Licentiate thesis, Division of Automatic Control, Department of Electrical Engineering, Linköping University, Sweden, 2003.
- [93] K.J. Beven, D.T. Leedal, P.J. Smith, and P.C. Young. Identification and representation of state dependent non-linearities in flood forecasting using the DBM methodology. In H. Garnier and L. Wang, editors, *System Identification, Environmental Modelling, and Control System Design*, pages 341–366. Springer, 2012.
- [94] P.C. Young. Optimal identification and estimation of discrete-time transfer function models. In *Recursive Estimation and Time-Series Analysis*. Springer, 2011.
- [95] R. Tóth. *Modeling and identification of linear parameter-varying systems*, volume 214. Springer, 2010.
- [96] A. Chiuso. The role of vector autoregressive modeling in predictor-based subspace identification. *Automatica*, 43(6):1034–1048, 2007.
- [97] W.E. Larimore. Canonical Variate Analysis in identification, filtering, and adaptive control. In *Proceedings of the 29th IEEE Conference on Decision and Control, 1990*, pages 596–604.
- [98] T. Kailath. *Linear systems*. Prentice-Hall Englewood Cliffs, NJ, 1980.
- [99] A. Alenany, H. Shang, M. Soliman, and I. Ziedan. Improved subspace identification with prior information using constrained least squares. *Control Theory & Applications, IET*, 5(13):1568–1576, 2011.
- [100] P. Trnka and V. Havlena. Subspace like identification incorporating prior information. *Automatica*, 45(4):1086–1091, 2009.
- [101] H. Garnier, P. Sibille, and A. Richard. Continuous-time canonical state-space model identification via Poisson Moment Functionals. In *Proceedings of the 34th IEEE Conference on Decision and Control*, volume 3, pages 3004–3009. IEEE, 1995.

- [102] G.C. Calafiore. Random convex programs: Dealing with the non-existing solution nuisance. In *49th Conference on Decision and Control (CDC)*, pages 1939–1944. IEEE, 2010.
- [103] L. Deori, S. Garatti, and M. Prandini. Stochastic constrained control: trading performance for state constraint feasibility. In *European Control Conference (ECC)*, pages 2740–2745. IEEE, 2013.
- [104] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.
- [105] A. Bemporad, F. Borrelli, and M. Morari. Min-max control of constrained uncertain discrete-time linear systems. *IEEE Transactions on Automatic Control*, 48(9):1600–1606, 2003.
- [106] A. Bemporad and M. Morari. Robust model predictive control: A survey. In *Robustness in identification and control*, pages 207–226. Springer, 1999.
- [107] D. la Pena, D. Munoz, T. Alamo, and A. Bemporad. A decomposition algorithm for feedback min-max model predictive control. In *Conference on Decision and Control and European Control Conference, CDC-ECC-05*, pages 5126–5131. IEEE, 2005.
- [108] M.V. Kothare, V. Balakrishnan, and M. Morari. Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32(10):1361–1379, 1996.
- [109] D.M. Raimondo, D. Limon, M. Lazar, L. Magni, and E.F. Camacho. Min-max model predictive control of nonlinear systems: A unifying overview on stability. *European Journal of Control*, 15(1):5–21, 2009.
- [110] P. Scokaert and D. Mayne. Min-max feedback model predictive control for constrained linear systems. *IEEE Transactions on Automatic Control*, 43(8):1136–1142, 1998.
- [111] M. Cannon, B. Kouvaritakis, and X. Wu. Model predictive control for systems with stochastic multiplicative uncertainty and probabilistic constraints. *Automatica*, 45(1):167–172, 2009.

- [112] D. Van Hessem and O. Bosgra. Stochastic closed-loop model predictive control of continuous nonlinear chemical processes. *Journal of Process Control*, 16(3):225–241, 2006.
- [113] G.C. Calafiore and L. Fagiano. Robust model predictive control via scenario optimization. *IEEE Transactions on Automatic Control*, 58(1):219–224, 2013.
- [114] P.J. Goulart, E.C. Kerrigan, and J.M. Maciejowski. Optimization over state feedback policies for robust control with constraints. *Automatica*, 42(4):523–533, 2006.
- [115] J. Lofberg. Approximations of closed-loop minimax MPC. In *42nd Conference on Decision and Control (CDC)*, volume 2, pages 1438–1442. IEEE, 2003.
- [116] D.Q. Mayne, M.M. Seron, and S.V. Raković. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41(2):219–224, 2005.
- [117] M.C. Campi and S. Garatti. The exact feasibility of randomized solutions of uncertain convex programs. *SIAM Journal on Optimization*, 19(3):1211–1230, 2008.
- [118] T. Alamo, R. Tempo, A. Luque, and D.R. Ramirez. Randomized methods for design of uncertain systems: Sample complexity and sequential algorithms. *Automatica*, 52(0):160 – 172, 2015.
- [119] Michael G. and Stephen B. CVX: Matlab software for disciplined convex programming, version 2.0 beta. <http://cvxr.com/cvx>, September 2013.
- [120] H.A. Nasir and E. Weyer. Control of a river with uncertain inflows. *Australian Control Conference (to appear)*, 2014.
- [121] L. Wang. *Model predictive control system design and implementation using MATLAB®*. Springer Science & Business Media, 2009.
- [122] G. Schildbach, L. Fagiano, and M. Morari. Randomized solutions to convex programs with multiple chance constraints. *SIAM Journal on Optimization*, 23(4):2479–2501, 2013.

-
- [123] Y. Oishi. Polynomial-time algorithms for probabilistic solutions of parameter-dependent linear matrix inequalities. *Automatica*, 43(3):538 – 545, 2007.
- [124] T. Alamo, R. Tempo, and E.F. Camacho. Randomized strategies for probabilistic solutions of uncertain feasibility and optimization problems. *Automatic Control, IEEE Transactions on*, 54(11):2545–2559, Nov 2009.
- [125] A. Carè, S. Garatti, and M.C. Campi. FAST-Fast Algorithm for the Scenario Technique. *Operations Research*, 62(3):662–671, 2014.
- [126] R.T. Rockafellar and S. Uryasev. Optimization of conditional value-at-risk. *Journal of risk*, 2:21–42, 2000.
- [127] H.A. Nasir, A. Carè, and E. Weyer. A randomised approach to flood control using Value-at-Risk. In *54th Conference on Decision and Control (CDC)*. IEEE, 2015.
- [128] S. Garatti and M.C. Campi. Modulating robustness in control design: principles and algorithms. *IEEE Control System Magazine*, 33(2):36–51, 2013.
- [129] M.C. Campi and A. Carè. Random convex programs with L₁-regularization: sparsity and generalization. *SIAM Journal on Control and Optimization*, 51(5):3532–3557, 2013.
- [130] A. Doucet, N.D. Freitas, and N. Gordon. *An introduction to sequential Monte Carlo methods*. Springer, 2001.
- [131] L.D. Brown, T.T. Cai, and A. DasGupta. Interval estimation for a binomial proportion. *Statistical science*, pages 101–117, 2001.



Minerva Access is the Institutional Repository of The University of Melbourne

Author/s:

Nasir, Hasan Arshad

Title:

System identification and control of rivers

Date:

2016

Persistent Link:

<http://hdl.handle.net/11343/116673>

File Description:

System Identification and Control of Rivers

Terms and Conditions:

Terms and Conditions: Copyright in works deposited in Minerva Access is retained by the copyright owner. The work may not be altered without permission from the copyright owner. Readers may only download, print and save electronic copies of whole works for their own personal non-commercial use. Any use that exceeds these limits requires permission from the copyright owner. Attribution is essential when quoting or paraphrasing from these works.